RMIT UNIVERSITY

Efficient Key Management in Wireless Sensor Network Security

A thesis submitted in fulfilment of the requirements for the degree of Master of Engineering

Xinyang Zhang

B. Eng.

School of Electrical and Computer Engineering

College of Science Engineering and Health

RMIT University

June 2015

Declaration

I certify that except where due acknowledgement has been made, the work is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program; any editorial work, paid or unpaid, carried out by a third party is acknowledged; and, ethics procedures and guidelines have been followed.

Xinyang Zhang

June 30, 2015

Acknowledgments

Throughout my candidature, I have been very fortunate to get help and support from many people. I would especially like to express my gratitude and heartfelt thanks to my supervisor, Dr. Jidong Wang, for his excellent supervision and support in this research.

I would like to express my sincere appreciation to Dr. Peter John Redcliffe for his support and guidance.

I would like to thank my family members. Although they are so far from me, they always encourage me and support my study. I gratefully acknowledge my father and my mother for all they have given me.

Finally, I would like to thanks my dear friend Bingyu Yi, for her encouragement through the hardest times.

Contents

List of Figures

List of Tables

Abstract

Wireless sensor network is a multi-hop ad hoc network formed by a large number of low-cost micro-sensor nodes which communicate through radio channels. It is widely used in many areas in modern society and attracts a lot of attention from researchers. This research is on wireless sensor network security and it focuses on key management in hierarchical wireless sensor networks. Through literature review, the drawback and weakness of existing key management schemes are analyzed from various aspects including key establishment, key distribution, key update, authentication and node operation mechanism. Assessment criteria for key management scheme are proposed under different requirements and constraints of wireless sensor networks. The security criteria cover keying model, key distribution, key update, node operation and resilience. For cluster based hierarchical wireless sensor networks, an assistant node is introduced in a cluster to deal with the situation of cluster head compromise and to keep the member nodes securely staying in the network. With introduction of assistant nodes, a complete secure efficient hierarchical key management scheme (SEHKM) for wireless sensor network is proposed. The scheme supports three types of keys and the big improvement over existing key management schemes is on group key update, which is based on pseudo-random numbers and group Diffie-Hellman. The analysis and evaluation have shown that that SEHKM offers strong security with efficient operation from energy consumption point of view.

Key words: wireless sensor network, security, key management, assistant node management, SEHKM, key update

1. Introduction

Wireless sensor network is a multi-hop ad hoc network formed by a large number of low-cost micro-sensor nodes which communicate through radio channels [1, 2]. It is widely used in many areas, such as military, agriculture, weather conditions and health care [3]. For applications in which, information protection is essential, security for the wireless sensor network needs to be considered. The topic has attracted a lot of attention from the researchers in wireless sensor network. There are many aspects in wireless sensor network security and key management is one of them. Key management includes key establishment, distribution and update. This research is focuses on key management. In this chapter, a background introduction of key management is presented. It also outlines significance, objective and scope of the research. Finally, the main structure of the thesis is outlined.

1.1. Background of Wireless Sensor Networks

In early 1970s, point to point transmission based sensor networks appeared. This is the first generation of sensor networks. With development in the related disciplines, there are many different types of sensors coming up. The processing capability was dramatically enhanced with advanced microprocessors and signal processing technology. This can be classified as the second generation of sensor networks. 1990s had seen large multi-functional intelligent sensors for various applications [4, 5]. With fast development of wireless and network technologies, radio based sensor networks became a trend in the field of sensor networks. Now wireless sensor networks have been widely used in many areas. Figure 1 shows a generally wireless sensor network. The main advantages of wireless sensor network are low cost and easy deployment and low operations maintenance. Sensor nodes can be randomly deployed in a designated area and self-organized in the initialization phase. For applications in dangerous environments such as volcano monitoring [6], it is too dangerous for human to attend at the sites and wireless sensor network becomes a good choice. A wireless sensor network's self-organization in its lifetime is a very important feature that reduces the human intervention in wireless sensor networks operation and maintenance [7]. The modern microprocessor technology and radio communication technology have made the low-cost wireless sensor network nodes available. That contributes to the wide wireless sensor network applications in many areas including military, agriculture, environment and health care. In military, wireless sensor networks are essential in C4ISRT (command, control, communication, computing, intelligence, surveillance, reconnaissance and targeting) system [8]. Wireless sensor networks have the advantages of fast deployment, self-organization and fault tolerance which lead to a promising future of wireless sensor networks in modern military technology. Because a wireless sensor network is formed of large number of low-cost sensor nodes, the network can survive even some of the nodes are damaged

or destroyed by the enemies. They can be utilized in battle field surveillance, reconnaissance of enemy forces and terrain, circumstance assessment of loss, targeting etc [9]. There are many existing successful military wireless sensor networks applications such as boomerang shooter detection system [10] and VigiNet [11]. Wireless sensor networks are also widely applied in environment and agriculture monitoring. The applications include animal tracing, environment condition monitoring, meteorological monitoring and disaster detection. The well-known GDI(Great Duck Island) [12, 13] which is a study on birds and CORIE [14], an environment monitoring system, are two successful application cases. The applications of wireless sensor networks in medical system and health care include monitoring and collection of body biology data, tracing doctors and patients in hospital and medicine management. CodeBlue [15], a medical care and disease response system, developed by Harvard University is a good example. Wireless sensor networks can also find applications in many other areas such as smart home, industrial automation, telecommunication etc [7]. Even with its wide application, wireless sensor network is still a young technology. Its advancing is a big challenge for the research community. Figure 1 shows a general wireless sensor network.

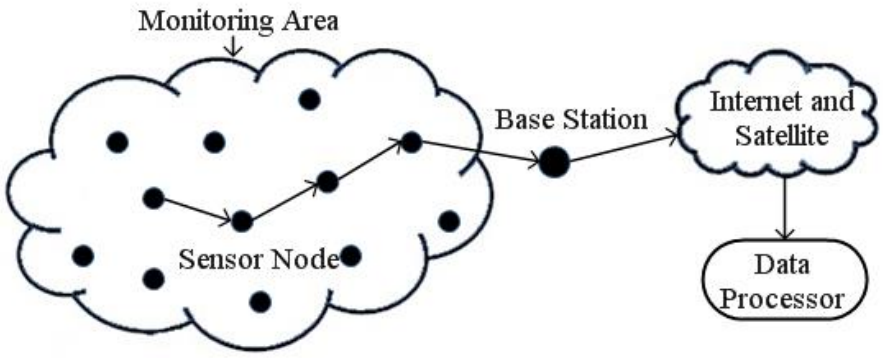

Figure 1: A wireless sensor network

1.1.1. Constraints of Wireless Sensor Networks

The wireless sensor network is formed by sensor nodes which are able to collect information or control end device. A sensor node built on an embedded system has three functions: sensor interface, data processing and network interface [7]. Sensor interface collects the sensed information. Some simple data processing functions can be carried out on the sensor nodes. The network interface is for communication and networking. A node usually has four units: a transceiver unit, a processing unit, a sensing unit and a power unit [16, 17]. The transceiver unit transmits and receives data and it links a node to a network. The processing unit includes processor and memory. It processes and stores data. The sensing unit has a sensor and analog to digital converter (ADC). It catches physical information from surrounding environment and converts information to digital data. The power unit usually is a battery which provides energy for all functions of a sensor node. Figure 2 describes a sensor node's units.

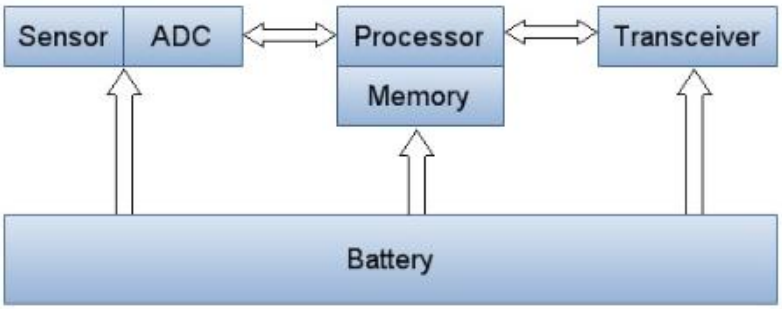

Figure 2: The components of a sensor node

In a wireless sensor network, sensor nodes are independent tiny devices. Each of them has its own individual battery. The general wireless sensor network constraints are listed in the following [18-20].

- **Hardware limitation:** As mentioned above, a number of functions need to be carried out in a node. Due to a node's small physical size and the low cost, its memory and processor power are limited. The computation and memory constraints need to be considered in wireless sensor network protocol and system design.
- **Limited energy supply:** wireless sensor network nodes are powered by battery. Once a node is deployed, the battery is not replaceable in many applications. In another words, a node's life is mainly dependent on its battery. The battery has to serve the lifetime of a node. The main consumers of the energy in a node are its data processing unit and network communication unit. Network design needs to optimize its overall performance with the constraints of limited energy supply to maximize wireless sensor network nodes' life.
- **Communication:** In a wireless sensor network, all communications are through radio channels. A node's radio transmission range should not be too big with power consumption consideration. i.e. node to node hop distance should be reasonable long. Radio channels suffer interference from the environment and other wireless communications. The data transmission is unreliable due to data error and packet loss. Radio signal is exposed and that means a radio receiving device in the range can intercept the radio signal. Without the security protection, wireless sensor network is vulnerable to information leak and various attacks.

1.1.2. Unique Characteristics of Wireless Sensor Networks

The special characteristics of wireless sensor network cannot be found in traditional networks. The understanding of these unique properties is helpful in wireless sensor network research. The unique characteristics of wireless sensor networks are listed below [21].

- **Limited resources:** As it is presented in last section, wireless sensor nodes

have limitation on their capabilities of computing, communication and energy supply.

- **Dynamic:** A sensor node should be removed from the network if its battery has exhausted or it has other failures. In some applications, a sensor node cannot serve the whole life of the network, thus some new nodes are required to be added into the network. This may leads to changed network topology which require dynamic functions.
- **Self-organization:** A wireless sensor network can be self-organized. It should be able to adjust to changes in condition in forming an independent network without human intervention.
- **Multi-hop communication:** In a wireless sensor network, a node is only capable to communicate with its immediate neighbor nodes. The communication of two nodes which are far away has to transmit data through several intermediate nodes. The sensor nodes in wireless sensor networks are data collector, sender and router at the same time.
- **Application relevance:** The characteristics of a wireless sensor network are highly dependent on its application. Different applications collect different kinds of data which makes them have differences in network topology, routing protocol, physical signal and etc. This is a huge difference with traditional networks.

1.1.3. Security in Wireless Sensor Networks

Security is an important factor to be considered in many wireless sensor network applications. Wireless sensor networks are even more vulnerable to attacks due to their unique characteristics. Listed below are the potential attacks on wireless sensor networks [22, 23].

- **Attacks against privacy:** In Wireless sensor networks, large volumes of information can be easily available through radio and remote access and this can compromise privacy. Attacks against privacy include traffic analysis, camouflages adversaries and monitor and eavesdropping.
- **Physical attacks:** Attackers can physically capture nodes deployed in hostile outdoor field. They can extract cryptographic keys, modify programming or replace the nodes with malicious nodes.
- **Node attacks:** These attacks act on node. They are node subversion, node malfunction, node outage, node replication and false node.
- **Passive information gathering:** An attacker can collect data from a wireless sensor network with enough resources and powerful hardware if data is not or poorly encrypted.
- **Attacks on information in transit:** wireless sensor network communication can be easily eavesdropped. Attackers can monitor the communication, interrupt, intercept, modify or fabricate packets thus, provide wrong information to the base stations or sinks at any time.

- **DoS:** This DoS attack attempts to exhaust resources on aim node by sending many useless messages and prevents user from accessing services or resources. In wireless sensor networks, DoS attacks at physical layer can be jamming and tampering. Collision and exhaustion unfairness are at link layer. Black holes, neglect and greed, misdirection are at network layer. Flooding and resynchronization are at transport layer.
- **Message corruption:** The integrity of a message modified by an attacker is compromised.
- **Routing attacks:** These are attacks happen on network layer and includes Sybil attack, sinkhole attack, wormhole attack, hello flood attack, selective forwarding and spoofed, altered and replayed routing information.

With various potential threats, wireless sensor networks need to have security mechanism to provide required security services for their applications. Wireless sensor network security requirements are summarized in the following [18, 23, 24].

- **Confidentiality:** Nodes should not reveal any data to unintended recipients.
- **Integrity:** Data should not be changed between transmissions due to the environment or malicious activities.
- **Data freshness:** Old data should not be used as new (i.e., prevent replay attacks).
- **Authentication:** Data used in decision-making processes must originate from the correct source.
- **Robustness:** When some nodes are compromised, the entire network should not also become compromised. The quantitative value with which this requirement should be satisfied depends on the application.
- **Self-organization:** Nodes should be independent and flexible enough to be self-organizing (autonomous) and self-healing (failure tolerant).
- **Availability:** The network should not fail frequently.
- **Time synchronization:** Collaborative node applications need time synchronization. Time synchronization protocols should not be manipulated to produce inaccurate time.
- **Secure localization:** Nodes should be able to accurately and securely acquire location information.

There are some security mechanisms provided in industrial wireless sensor network standards, such as ZigBee [25]. However, wireless sensor network security is far from mature. There are many ongoing research works in various aspects of wireless sensor networks. On service level, there are works on secure data aggregation, secure localization and intrusion detection. At the network level, node authentication and secure routing are required to protect nodes in a network from being connected to unauthorized parties and secure its communication with the base station. The main

security mechanism is based on encryption and authentication. When a node transmits data to another node, it needs encryption to provide data confidentiality. When a node receives data from its neighbor, it needs to authenticate the data is genuinely from that node. For encryption and authentication to work, a security framework and security key management is needed. A general wireless sensor network security mechanism and service diagram [26] is shown in Figure 3. Wireless sensor network security key management is one of the hot wireless sensor network research areas and it is the focus of this research.

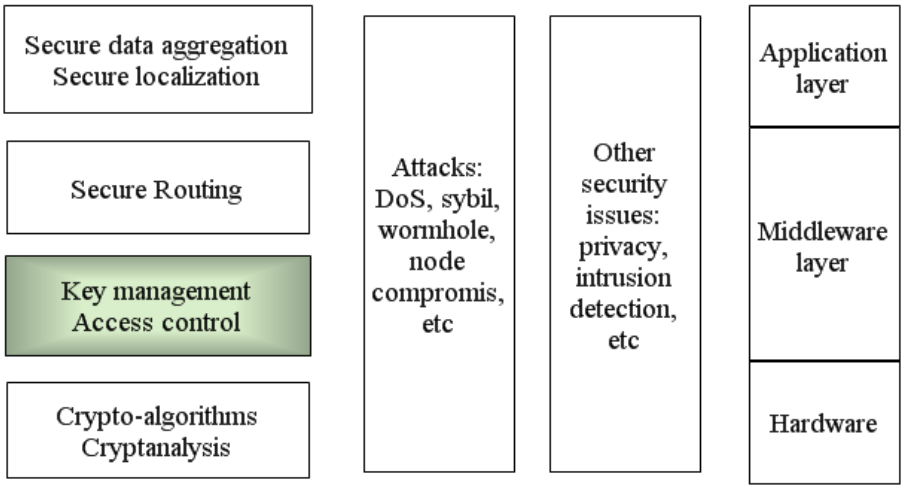

Figure 3: Security mechanism and srvices in wireless sensor networks

1.2. Motivation of the Research

Wireless sensor network has found wide applications in many areas, such as battlefield surveillance, environmental monitoring, health care, target tracking and so on [7]. For some applications, security is a very important issue. For example, in some critical infrastructure (such as power plant, smart grid, big dam) monitoring and control systems, if the security is compromised, the result will be devastating, especially in today's world with big threat from all kinds of terrorists. All these reasons have made the wireless sensor network security a significant research area. Key management can be considered as the core of wireless sensor network security.

Wireless sensor network security key management includes the security structure, key generation, key distribution and key refreshment. Due to wireless sensor networks' many constraints in computation, communication, memory and energy, key management schemes need to have good power efficiency apart from providing strong security.

Different wireless sensor network applications have different features and different security requirements. For example, volcano monitoring network and military surveillance network are supposed to use different hardware and have different

security requirement. The latter is more sensible to information leaking and enemy attack. Generally, security strength is only one factor in the overall wireless sensor network design. With the constraints discussed previously, network performance, node life and security have to be considered all together. Some applications may have more emphasis on performance and for others security may be the number one consideration. In most cases, long nodes' life is expected. Therefore, there will be compromise among all aspects of a wireless sensor network. For example, if a wireless sensor network provides a very strong security, but only lasts a very short period of time, then its application value is limited. This research is to search for good key management schemes, not only offering strong security, but also having good power efficiency.

This research is significant. Firstly, the proposed key management scheme combines two existing schemes and has advantages over them. It applies secure key management in HWireless sensor networks. Moreover, the proposed scheme is efficient and secure. It reduces resources cost while maintain security. In addition the scheme is flexible, scalable and accessible. Thus the scheme is suitable for large scale Wireless sensor networks.

1.3. Objective and Scope of the Research

Hierarchical structured wireless sensor network is a very popular category of wireless sensor networks. It offers good features on network operation, data aggregation and security. This research focuses on key management in hierarchical wireless sensor networks. Based on existing schemes, new key management scheme is proposed for a hierarchical wireless sensor network. The main goal is on the improvement of power efficiency and security. Other requirements, such as the flexibility, scalability and accessibility are also considered. The main tasks of this research are as follows:

- To analyze the requirements of wireless sensor network key management and existing problems. The main key management schemes are assessed on security, efficiency and operation requirements. The features of each scheme are outlined.
- To propose a new key management scheme based on combination of two existing schemes. It includes key establishment, key distribution and key update. The proposed scheme should satisfy as many requirements of key management as possible.
- To analyze and evaluate the newly proposed key management scheme on security strength and performance. Simulation results are required to back the analysis.

1.4. Structure of Thesis

The structure of this thesis is as follow:

- **Chapter 1**: **Introduction.** An introduction of wireless sensor networks and wireless sensor network security background is given. Then the motivation, objective and scope of the research are presented. The structure of thesis is outlined.
- **Chapter 2: Review of wireless sensor networks key management schemes.** In this chapter, requirements of key management in wireless sensor networks are discussed. Three simple keying models are described. Classification of key management schemes is presented. Some popular key management schemes are reviewed. Finally, based on the above analysis, research questions are proposed.
- **Chapter 3: The criteria for key management scheme assessment.** This chapter is the answer of first research question. Firstly, the requirements and contents of key management are analyzed. Criteria to assess security of key management schemes are proposed. Also the criteria on efficiency and operation are given as well. Finally the assessments of two existing schemes based on the proposed criteria are discussed.
- **Chapter 4: Cluster based wireless sensor networks and key management.** This chapter includes the answer of second research question. Firstly, the architecture of wireless sensor networks and node clustering are introduced. Then the definition and operation of assistant node are described. The security analysis of the assistant node is presented too.
- **Chapter 5: The secure efficient hierarchical key management scheme.** This chapter presents the proposed scheme and it includes the answer of last research question. Firstly, two popular key management schemes are further analyzed. Then, the proposed scheme is presented to address the issues in the existing schemes. The proposed scheme supports three types of keys and the key establishment, distribution and update are described. The key management in node operation is shown too.
- **Chapter 6**: **The evaluation of SEHKM.** The security strength and the performance of the proposed scheme is analyzed and discussed. The simulation results are presented to verify the scheme's improvement.
- **Chapter 7: Conclusion.** In this chapter, the contribution of research is highlighted. The future works are discussed.

1.5. Summary

Wireless sensor network is a popular technology which is widely applied in many areas. Security as an important issue has attracted a lot of attention of researchers. This research focuses on key management for wireless sensor network security. In this chapter, the background information of the research topic is introduced. The motivation, significance and objective of research are presented. Lastly, the structure of the thesis is outlined. In the next chapter, the requirements and classification of key management scheme in wireless sensor networks will be discussed. Several existing schemes will be analyzed.

2. Review of Wireless Sensor Network Key Management Schemes

Key management is a big challenging issue in wireless sensor network as it forms the basic security structure of the networks. Researchers attempt to apply existing security models and techniques of traditional networks on wireless sensor network. But there is a huge difference between sensor nodes and computers and they have different requirements and constraints. Existing security methods in traditional networks are not suitable on wireless sensor networks and they need to be changed. Various wireless sensor network key management schemes proposed so far attempt to satisfy the security, efficiency and operation requirements under the resource constrain of wireless sensor network.

2.1. Requirements of Key Management in Wireless Sensor Networks

As a main aspect of the wireless sensor network security, key management attempted to achieve some of security requirements [18, 23, 24] mentioned in section 1.1.3. The first is confidentiality. It is that the content of the information flowing in wireless sensor network must be protected from disclosure to unauthorized parties. The second is authentication which is the parties who are able to access to the shared information should be identified and authenticated. The third one is data integrity and it means data should not be changed between transmissions due to the environment or malicious activities. Robustness is another requirement which deals with node compromise and attack. Freshness requirement in key management scheme should guarantee that each node gets the updated shared key. It should include forward and backward compatibility (or secrecy). Forward compatibility requires that a node should not be able to access the network and conduct further communication after it is compromised or departed. Backward compatibility is that a newly joined node should be able to access the network and communicate with other nodes, but not able to access stored previous communications [25].

Security is not the only requirement of wireless sensor network key management. The architecture of wireless sensor network makes it different from traditional network and requires more [25]. First, nodes should be replaceable when compromised, damaged or out of power. It should be flexible and it requires that On-the-fly addition and revocation of nodes should also be supported. Moreover, scalability is another consideration. Wireless sensor network security key management schemes should operate on different level of security requirement and different size of wireless sensor networks. Besides, accessibility which addresses node

to node communications in the network adds another dimension in the requirement matrix. Flexibility, scalability and accessibility are the operation requirement of wireless sensor network.

According to the constraints of wireless sensor network [18], key management schemes in wireless sensor networks should also consider efficiency requirements which include communication, computation, storage, and power consumption. These are wireless sensor network specific issues. The key management mechanism employed in a wireless sensor network should provide not only the required security strength, but also efficiency under the limited resources and operation requirement. Table 1 shows all the requirements of wireless sensor network key management.

Table 1: Requirements of key management in wireless sensor network security

Requirements	Details
Security requirement	Confidentiality, authentication, data integrity, freshness and robustness.
Efficiency requirement	Reasonable communication, computation, storage and power consumption.
Operation requirement	Flexibility, scalability and accessibility

2.2. Keying Models in Wireless Sensor Network

There are three types of key models based on communication relationships. They are network key, pairwise key and group key [27]. Table 2 shows their characteristics.

Table 2: Three types of keying models

Model	Description	Advantage	Disadvantage
Network	The whole network shared one key.	Good efficiency, accessibility, flexibility and scalability.	Security is weak.
Pairwise	Each pair of nodes shared one key, different pairs have different keys.	Good accessibility and strong security strength.	Efficiency, scalability and flexibility are not good.
Group	Each group of nodes shared one key, different groups use different keys.	It combines the network and pairwise key model, has better security than network key and better efficiency than pairwise key. Accessibility, flexibility and scalability are good.	Not easy to implement. Not suitable for all networks with different structure.

These three keying models are highlighted here because all key management schemes are combined by them. They are basic element for a key management.

2.3. Classification of Key Management Schemes

The existing wireless sensor network key management protocols can be classified in many ways. Here are several classifications based on different conditions.

2.3.1. Single and Mixed Model Key Management Schemes

The difference between single and mixed schemes is how many basic keying models the scheme uses. Single model key management applies only one kind of keying model while mixed model scheme applies a hybrid approach. Their advantages and disadvantages are compared in Table 2. Generally, the mixed model schemes have better security and worse efficiency than the single model ones.

2.3.2. Hierarchical and Flat Wireless Sensor Network Key Management Schemes

Wireless sensor networks can be classified as hierarchical networks and flat networks by network structure [1]. Figure 4 shows the hierarchical network and flat network.

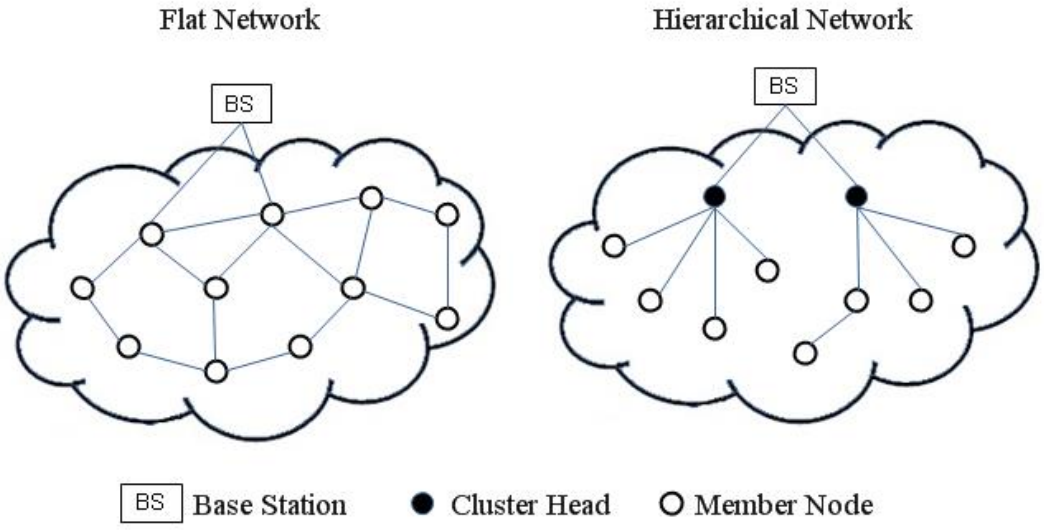

Figure 4: Wireless sensor network topologies

In a hierarchical wireless sensor network, all nodes are divided into several clusters and each cluster has a head. A cluster head leads a group of general sensor nodes and all cluster heads are led by the head of cluster head. Every node belongs to a cluster and communicates with base station through cluster head. In a flat network, all nodes have the same capability; they communicate with neighbors and transmit data to base

station one by one. Comparing with flat wireless sensor network, hierarchical wireless sensor network has many advantages. First, in hierarchical network, cluster head and base station manage the network. Normal nodes can only wake up when they are needed for data transmission or collection. Hence, the energy consumption would be reduced for this. Then, cluster head and cluster members exchange information in a cluster, it helps cluster head concludes the local information. Lastly, there is less competition between nodes for communication channel as cluster head transmits most data, more nodes can be deployed in the network. The hierarchical wireless sensor network has better scalability. However a cluster head manages the keys of all nodes in the same cluster. Hence the cluster will face serious problems if the head is compromised or damaged. This research focuses on key management in hierarchical wireless sensor network security.

Eschenauer is a flat wireless sensor network key management. In this kind of schemes [21, 28-33], all nodes have the same ability and they manage the keys themselves or by working together. So the costs on each node are almost the same and it is higher than the normal nodes in a hierarchical wireless sensor network.

2.3.3. Static and Dynamic Key Management Schemes

According to whether or not the key management schemes consider the key refresh and node addition, replacement and revocation, they can be classified as static and dynamic key management [27]. The static schemes [21, 34] have no changes after initialization phase, do not refresh the keys and the network scale does not change, so there is no cost for these. However if the node is compromised, it leads to security problems. But the attackers hardly get the keys by compromising a node in a wireless sensor network which employ dynamic key management [35] because the keys change dynamically all the time. The dynamic schemes provide higher security strength, but it cost more computation, energy and communication resources.

2.3.4. Stochastic and Deterministic Key Management Schemes

In a stochastic key management scheme [28, 29], nodes take keys from the key pool randomly. The key distribution is simple and cost less computation and energy. But the nodes may store much more keys than they need and this kind of schemes may have problems on accessibility.

The keys generated in a determinate way in a deterministic scheme [34]. Such as by some algorithms or identified information. These schemes need more computation, power and communication resources than the stochastic ones. But the memory usage is less and the nodes are able to access each other.

2.3.5. Symmetric and Asymmetric Key Management Schemes

The difference of the two type's schemes is the encryption algorithm they used. The symmetric algorithm [28, 34] encrypts and decrypts messages by the same key. The usages of energy, communication and computation are small. But an asymmetric scheme [36] is securer because it uses different keys in encryption and decryption. It requires more resources. Figure 5 shows symmetric algorithm and asymmetric algorithm.

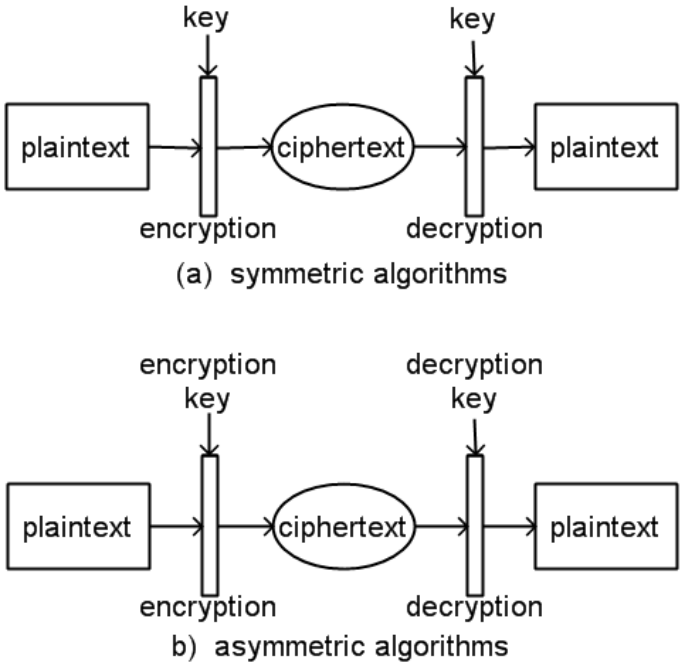

Figure 5: (a) symmetric algorithm (b) asymmetric algorithm

As the result of constraints of wireless sensor network, symmetric algorithm is more suitable for it. Because asymmetric key management costs more computation and energy resources, it is rarely applied on wireless sensor network. But there are some researches show that improved asymmetric algorithm can be applied on wireless sensor network. If the problem of resource limitation can be solved, asymmetric key management can be more attractive as its higher security level. There is a mechanism mixed the symmetric and asymmetric encryption, but it costs a lot and is for heterogeneous wireless sensor network.

2.4. Popular Key Management Schemes

2.4.1. Eschenauer Scheme

Eschenauer [28] key pre-distribution scheme for wireless sensor network is one of the early key management schemes for wireless sensor network. In this scheme, a large key pool is generated for a wireless sensor network with N nodes. Each sensor node randomly takes k keys from the pool to establish a key ring where $1<k<<N$. So every node has a unique key ring which contains k keys. When two nodes need to

communicate, they broadcast their keys' IDs and search a common key in the key rings. If there is no common key, they find the third node which has the common keys with the both nodes which means the third one is able to communicate with the two nodes. It is possible for the two nodes to talk to each other by the third party.

It is obviously that this scheme has the advantage that the size of key pool and key rings are scalable. In real applications the key pool and key rings can be adjusted appropriately when the networks expand or shrink. However, Eschenauer scheme fails to mention about authentication which is a requirement of wireless sensor network security. In addition, there is no cluster operation in the scheme. So it cannot be applied in hierarchical wireless sensor network which is an important structure of wireless sensor network. Then, Eschenauer may cost more energy to satisfy accessibility because some communication between nodes may pass many others. Lastly, when the size of network is big, the small key pool leads to weak security strength and large key pool will increase the memory cost.

2.4.2. Du Scheme

Du [29] proposed a key management scheme based on Eschenauer's and Blom's [37] in 2003. It uses the Blom's key matrix instead of individual keys in Eschenauer scheme.

Blom's model is based on the idea of a symmetric matrix multiplication, where row i column j is equivalent to row j column i. Thus, when node i calculates key ij and node j calculates key ji, the keys are identical, leading to a commonly shared secret. Blom's scheme distributes the information required for this calculation in terms of a public matrix and a private matrix.

It generates a pool of key matrices, and then each node take a subset of the key matrix pool instead of key ring in Eschenauer. When two nodes establish communication, they find the common key matrix and calculate the pairwise key by Blom's scheme. If there is no common key matrix, they find a third party which has common key matrices with the both nodes.

The benefit of Du scheme is the strong robustness against node compromise. Because Du is similar to Eschenauer, the schemes have some common problems. It lacks the authentication and cluster operations. The accessibility is hard to be satisfied and it may cost more energy. On the other hand, it uses Blom's scheme and it makes the proposed scheme very complex. It is difficult to implement in real sensor networks and the complexity also led to high energy consumption.

2.4.3. Panja Scheme

Based on TDGH [38], a Tree based Group Deffi-Hellman algorithm, Panja [39] proposed a key management mechanism which is suitable for hierarchical wireless sensor network. The structure is a cluster head leads a group of general sensor nodes and all cluster heads are led by the head of cluster head. All sensor nodes pre-install a symmetric key for key transport encryption. Every node has a partial key to generate group keys. The leaf nodes generate random numbers to calculate their partial keys. The partial keys of parent nodes are calculated by the partial keys of the leaf nodes. There are two types of group keys: intra-cluster key and inter-cluster key. The intra-cluster group key is used for encryption/decryption of messages inside a sensor network group, whereas the inter-cluster group key is used for groups of cluster heads. A cluster head computes the intra-cluster group key by using all the partial keys in the group as arguments. The inter-cluster group key is generated in the same approach by using the partial keys of cluster heads.

This scheme is simple and easy to implement. The energy consumption is reasonable as Panja analyzed. If the size of cluster and partial key is small, the memory usage and computation costs can be reduced. It also satisfies the requirement scalability, accessibility and flexibility. But the scheme has weaknesses too. In Panja the group size decides the key strength, if the group is small, the key would be weak. But if the group size or partial key is bigger, there would be more computation and memory cost. Another drawback of Panja is the scheme only supports one type of key. If one node is compromised, the key which shared by the group of nodes would be disclosed and all nodes in the same group are in danger.

2.4.4. LEAP Scheme

LEAP (Localized Encryption and Authentication Protocol) [34] is a key management protocol with the operation of four types of keys: individual key, pairwise key, cluster key and group key. The individual key is for each node shared with base station. A pairwise key shared with another node. The cluster key shared with a node and all its neighbors. And group key is for base station communicates with all other sensor nodes in the network. These four keys are generated by a pre-distributed key called initial key. Firstly, the individual key is calculated by a function with the ID of the node. Secondly, nodes broadcast their IDs in the neighbor discovering phase and the receiver uses a function with initial key to establish the pairwise key shared with the neighbor. Then all nodes delete the initial key after the pairwise key generation phase. Next is cluster key distribution. A node secures the key with the pairwise key and broadcasts it to all neighbors. Lastly, the base station broadcasts the group key cluster by cluster.

LEAP uses μTimed Efficient Streaming Loss-tolerant Authentication Protocol (μTESLA) [40] to authenticate the broadcast of base station which make sure that

packet with group key is just sent by base station. The authentication for cluster key is a scheme called one-way hash-key chain.

LEAP has many advantages. It provides security services for different types of packet. It also meets the requirement of accessibility. However LEAP failed to satisfy the flexibility requirement because it assumes a static network. There is no node addition in this scheme. In addition, if initial key is disclosed, the adversary will be able to establish pairwise key with any node in the network.

2.4.5. Time-based Key Management Scheme

Jang [41] tried to improve LEAP and proposed a scheme which splits the network lifetime into P slots and every slot has an initial key. All nodes deployed in the same slot i are in the group Ni, they have the same initial key IKi and n master keys of n time slots which are chosen randomly. The master key helps nodes to calculate the initial keys of nodes in different slots. Then based on LEAP, the individual key, pairwise key, cluster key and group key can be generated. Figure 6 shows key materials pre-loaded to nodes at different time slots in the scheme

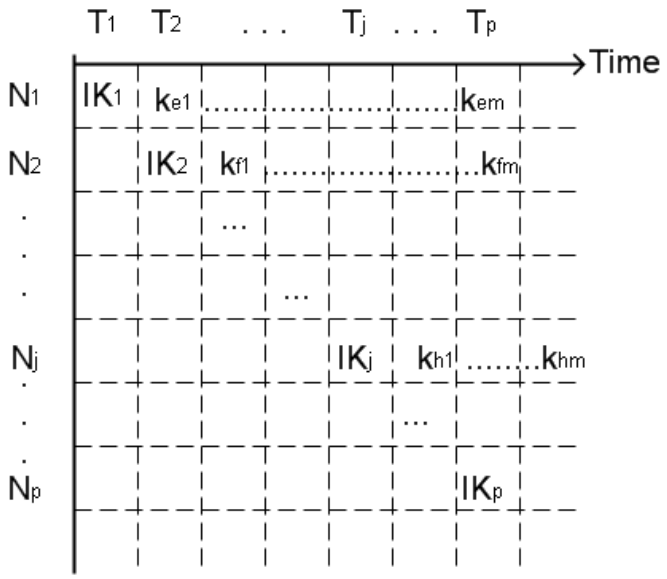

Figure 6: Key materials pre-loaded to nodes at different time slots in the scheme

Time-based scheme has a stronger initial key than LEAP because it uses different ones in different slots. But the master keys in one group are chosen randomly, it led to cost of computation and energy because when a node attempts to communicate with another one in different slot, it may go through many other nodes in different slots. Furthermore Jang did not solve the flexibility problem of LEAP.

2.4.6. TLA Scheme

Maala [42] proposed a key management scheme with two-level structure. It combined the key management mechanisms in each level. The techniques are: master key technique, pairwise key technique and key pool technique. Master key is a unique key which is pre-distributed and shared by all nodes in the network. Pairwise key is pre-distributed and shared by a pair of nodes. Each pair of nodes shared a different key. The key pool technique is similar with Eschenauer that each node randomly chooses and stores a subset of keys form the key pool as their key rings, then two nodes secure their communication by the same key in their key rings. The scheme divides the wireless sensor network into two types of clusters: supervised and unsupervised clusters. The supervised cluster has a supervisor node and a group of supervised nodes, while the unsupervised cluster only has unsupervised nodes without security requirements. The combined master and pairwise key technique is used in the supervised cluster while the key pool technique is used in unsupervised cluster.

The advantage of TLA is its capability against node capture. In addition the memory cost is reasonable. But when network expend and has more sensor nodes, the power consumption increases. It does not mention the nodes addition and revocation too. Lastly, nodes should be replaced after captured, however it is not covered in the scheme.

2.4.7. ECDH for Zigbee Pro

This mechanism [36] applies and improves the elliptic curve Diffie-Hellman (ECDH) key distribution scheme [43] for Zigbee Pro [44]. Zigbee Pro is a popular industrial standard of wireless sensor networks. It has three types of keys: master key, network key and link key. Master key authenticates the new node, generates link keys in application layer. Network key is shared by all nodes for broadcast. Master key and network key are pre- installed. Link key is pairwise key and uses for end-to-end encryption. It could be pre-installed or generated by master key.

ECDH for Zigbee Pro improved security level. It provides confidentiality, authentication and integrity. It also prevents man-in-the-middle attack using subMAC and replay attack. The mechanism also provides the almost same efficiency with Zigbee Pro. But, this scheme can only be used on Zigbee based wireless sensor network.

2.4.8. Research Questions

Based on analysis of current research situation and existing schemes, some of the problems are found.

1. Lack of criteria for key management scheme evaluation, especially on security. Most schemes analyze security by descriptive words in paragraphs which are

hard to compare with other schemes. It is also difficult to assess whether a scheme is good or not.

2. There is no method to solve problems caused by cluster head compromise. In hierarchical wireless sensor networks, it is impossible for base station to control all nodes in the network. This leads to significance of cluster head in efficient control of the whole network. However the importance of the cluster head makes it easy to be a target, and in most schemes, it is not mentioned that how to protect normal nodes after cluster head compromise.
3. High cost on security service of key update. Key update is an important method to keep secrecy of keys. To ensure keys are not disclosed after node replacement, revocation and compromise, the keys known by the node should be renewed. But key update includes key establishment and distribution which require lots of resources. As limitation of wireless sensor networks, costs of key update need to be reduced.

According to the above problems, the research questions are proposed. They are:

- How can the criteria be established to assess the security and efficiency of key management schemes?
- How can the security of other nodes in the same cluster be strengthened when the cluster head is compromised in hierarchical wireless sensor networks?
- How can the existing key update algorithm on hierarchical wireless sensor network be improved on efficiency?

2.5. Summary

In this chapter, a survey of existing key management schemes is presented and several popular schemes are analyzed. Firstly, the key management requirements which include security, efficiency and operation are introduced. Then the key management schemes are classified based on different characteristics of wireless sensor networks. After that, several popular schemes are introduced, the advantages and drawbacks of which are pointed out. Finally the common problems of current research are concluded, based on which the research questions are proposed.

3. The Criteria for Key Management Scheme Assessment

3.1. Introduction

Key management is important for wireless sensor networks and researchers proposed many schemes. In the last chapter, many key management schemes are introduced and analyzed. However it is hard for a user who is not in this area to assess a scheme. In order to facilitate users, key management scheme's preliminary criteria, especially security criteria are proposed in this chapter. Firstly, the constraints of wireless sensor networks and requirements of key management are summarized. Then there is an explanation of how key management works in a network, which includes initialization of keys, key generation and distribution, key management for node operations and actions in scenario of node compromise. Next the preliminary criteria of key management assessment are proposed. Security criteria include keying model, key distribution, key update, node operation and resilience. Efficient criteria include computation, communication and memory costs. The operation criteria are flexibility, scalability and accessibility.

3.2. The Requirement of Wireless Sensor Networks Key Management Schemes

As it is introduced in chapter 1 and 2, because of constraints of wireless sensor networks, the design of key management scheme has to consider security, efficiency and operation requirements. Thus key management scheme assessment also has to consider these three aspects. So there are three types of criteria: security, efficiency and operation criteria. Firstly, there are five aspects in security criterion: keying model, key distribution, key update, node operation, resilience. In the other hand, efficiency is easy to assess because the cost of energy and memory can be quantified. Simulation tools can evaluate schemes and calculation results are accurate data of energy and time consumption. At last, operation aspect should be considered as the capabilities of flexibility, scalability and accessibility.

3.3. Process of Key Management

3.3.1. Pre-distributed Keys

In order to provide efficient solutions to satisfy key management requirements, pre-distribution of keys is proposed which is loading keys into sensor nodes before deployment of nodes [18, 30, 45]. However, pre-distributed keying can cause problems due to its inflexibility in changing mission configurations. There are two types of popular used pre-distributed keying approaches.

3.3.1.1. Network-Wide Pre-distributed Keying

One of the simplest and most efficient approaches is network-wide pre-distributed keying [18]. Each node in the network is pre-loaded with the same keying material before deployment. Sometimes the number of keying material can be more than one which is depended on different requirements of application. Because each node in the network has the same keying material, keys for confidentiality and authentication can be generated conveniently with low cost on communication or computation.

However, network-wide pre-distributed keying may cause some problems. Firstly, it is not secure enough for some applications, for example, military scenarios. Enemies can easily get confidentiality keys that all nodes in the network employed by compromising only one node. The future and past recorded communications are potentially exposed to enemies.

3.3.1.2. Node-Specific Pre-distributed Keying

The node-specific pre-distributed keying [18] is the approach that the key shared by specific nodes is calculated and pre-loaded into these nodes before their deployment. Then these nodes are able to establish communication by the pre-distributed key. This approach provides strong security since the key is shared by specific nodes, thus adversary can only access these nodes, and other nodes in the network are still secured. Besides the approach costs very few energy on nodes.

But node-specific pre-distributed keying cannot satisfy flexibility and scalability. Because a deployed node only has the capability to communicate with nodes which share the same key with it, it is complex for people to arrange the position of each node during node deployment. If a node is surrounded by nodes sharing the different key, the node cannot join network due to disable of communication.

3.3.2. Key Establishment and Key Distribution

Establishing a cryptographic key between two or more participants requires two basic steps [18]. The first one is establishing trust between the participating entities. The second is computation of the cryptographic key. Both steps have unique requirements for maintaining key confidentiality, providing sufficient authentication and integrity protection, providing availability, etc.

In most key management schemes, the first step is done by the pre-installed keying materials. This is an efficient and secure approach to establish trust between nodes as it is presented in the last section. Security of the scheme without pre-distributed keying materials is critical as any untrusted entity is able to join the network. Some schemes employ public key cryptography (PKC) [46] based mechanisms to provide confidentiality and authentication. PKC based mechanism [47, 48] is asymmetric-key algorithms which the encryption key is different from decryption key. The main advantage of PKC based mechanism is strong security since each pair of public key

and private key is unique. But asymmetric algorithm always consumes much more resources than symmetric algorithm. The employment of PKC based mechanism should consider the resources costs on each node.

After trust establishment of participating entities, the cryptographic key can be computed. Some schemes generate pseudo-random numbers as keys due to convenience, but the security of the key is highly rely on randomness of pseudo-random number generation algorithm. Other schemes employ different algorithms to generate keys such as Diffie-Hellman.

Key distribution is to deliver keys from the key generator node to other nodes sharing the keys. It is the next step after key establishment. In some schemes, nodes do not distribute keys directly; instead, they distribute seeds to generate keys. This is an efficient approach to improve security of keys.

3.3.3. Key Management in Node Operation

In most wireless sensor networks, especially large scale and long lifetime applications, there are many situations require node operations. For example, a sensor node has to be replaced when its battery run out. And if a node is compromised, it has to be removed from the network. Thus key management schemes should be flexible in node addition, revocation and replacement. On the other hand, sensor nodes usually are deployed in outdoor field without surveillance, therefore they are easy to be captured by potential adversary. These compromised nodes should be removed from the network with consideration of network security. Key management scheme should support approach for node compromise too.

- **Node addition:** Node addition is adding new nodes into a deployed network. It is usually caused by expansion of the network or replacement of nodes with issues. Since only authorized nodes can join the network to maintain security, the new node has to pre-distribute at least some of the same keying material with the deployed nodes which helps it to be authenticated. After joining network, the newly joined node generates or receives other currently employed keys to establish communication with other nodes in the network.
- **Node revocation:** Node revocation is to remove nodes from a network. Once a node has problems such as power depletion, damage and compromise by adversary, the node has to be removed from the network for security and efficiency reasons. A node always contains confidentiality keys of the network which need to be updated after revocation of the node, to ensure backward secrecy of the network.
- **Node replacement:** Node replacement is to replace an old node with a new node due to problems of the old node such as energy exhaustion. Usually it includes addition of a new node and revocation of the old node. All keys that the replaced node has should be renewed after node replacement.

3.3.4. Key Update

In order to secure the forward and backward secrecy of wireless sensor networks, some potentially disclosed keys have to be replaced by new ones [49]. The process of establishment and distribution of new keys to renew the unsecured keys is key update. It usually happens after node revocation, replacement and compromise. In some applications that the wireless sensor networks deployed in a hostile environment such as military battle fields, some frequently used keys are possibly analyzed and calculated by enemies after sometime. In order to prevent it, these keys should be updated regularly to keep the secrecy. Key update is to establish and distribute a new key, so the resources cost is depended on key establishment and distribution algorithm. Besides, key update is different from key renew after node compromise which will be introduced in the next section, the aim of key update is to renew a key before adversary getting the key by analysis even the key is not compromised yet.

3.3.5. Node Compromise

Node compromise is an active attack which can cause confidentiality fail [1]. Generally, an adversary captures a node and digs out data in it. In another hand, the adversary is able to derive the data in a node by analyzing data got from other compromised nodes. The adversary can launch more attacks by controlling these compromised nodes. Thus the compromised nodes should be removed from the network as soon as possible. The keys that compromised nodes have are disclosed and cannot be used anymore. They need to be renewed in a safe way after detection of a compromised node. Intrusion detection [50] is essential during this process.

3.4. Criteria for Security Assessment

In current research, researchers evaluate their schemes by descriptions which can be not convincing but difficult to find by readers who are not in the area, especially normal users of wireless sensor network application. In order to solve the problem, preliminary criteria of security are established in this section by different process of key management to assess security strength of a key management scheme. The proposed criteria assess security of a scheme from keying model, key distribution, key update, node operation and resilience. Each of aspect is quantized as 20 points. The total score is 100 and the scheme has higher score is more secure than the one gets lower score.

3.4.1. Keying Model

Keying Models are introduced in section 2.2. They are pairwise keying model, group keying model and network keying model. Pairwise key provides encryption for messages between two specific nodes which cannot be decrypted by any others. Thus compromise this key by an adversary may lead to disclosure of communications between two nodes. In addition, a group key protects communication of a group of

nodes. Therefore if one node of the group is compromised, the communication of the whole group is potentially disclosing. Network key is one key shared by all nodes in the network. An adversary is able to eavesdrop on the whole network by compromising only one node. In conclusion, group keying model provides higher security strength than network keying model, but lower security than pairwise key.

In current research, a key management scheme usually employs at least one of the keying models. The scheme employs one keying model is a single model scheme while the scheme employs more than one keying model is a mix model scheme. In mixed model scheme, pairwise key is often employed by specific nodes and works with group key or network key. Conventionally, a mix model scheme is securer than a single model scheme. But single model scheme which employ pairwise key between each node has very strong security too. Table 3 shows the keying model criterion of key management.

Table 3: The keying model criterion

Keying model	Detail	Security score (20/20)	Conclusion of security strength
Single model scheme	Network key	5	Weak
	Group key	10	Normal
	Pairwise key	20	Very strong
Mixed model scheme	Network key and pairwise key	15	Strong
	Three types of key	20	Very strong

3.4.2. Key Distribution

Key distribution is to transmit keys from generator to other node sharing them which is potentially eavesdropping by adversaries. Thus the distributed keying materials should be encrypted to prevent them from being disclosed. A scheme also can employ algorithms such as Diffie-Hellman which is able to keep security even the keying material is eavesdropped.

As a criterion of security, a scheme should provide encryption for key distribution basically. Furthermore in situation of node compromise, a secure scheme provides encryption by a key which is not shared by the compromised node. Moreover, secret seeds exchange is a secure approach too. Table 4 shows the key distribution criterion.

Table 4: The key distribution criterion

Key distribution	Detail	Security score (20/20)	Conclusion of security strength
Directly key distribution	Basic encryption	5	Weak
	Encryption by different key	20	Very strong
Secret seed exchange	Exchange seeds which cannot calculate the key	20	Very strong

3.4.3. Key Update

Key update service protects the forward and backward secrecy of the network. In key update, the distribution of keys after node operations and compromise should be different for regular key update. In regular key update, new key can be encrypted by old key because it only prevent potential adversary from analyzing old key. However the aim of the specific key update after node revocation, replacement and compromise is to prevent the removed or compromised node from receiving following message by old keys. So the distribution of specific update has to be protected by a key besides old keys. Table 5 shows the key update criterion.

Table 5: The key update criterion

Key update	Detail	Security score (20/20)	Conclusion of security strength
Regular update	Just regular update	10	Normal
Specific update	Just specific update	10	Normal
Both approaches	Both regular and specific update	20	Very strong

3.4.4. Node Operation

It provides services to meet operation requirement of wireless sensor networks which should include keying services for node addition, revocation and replacement. Table 6 shows the node operation criterion.

3.4.5. Resilience

This is the resilience against node compromise which is the capability to keep secrecy of keys after one node compromise. If some keys are disclosed and the whole network is in danger after node compromise, the resilience is weak; if the compromised keys lead to potential danger to parts of nodes, the security is normal; and if the compromised keys can be update safely, the security is very strong. Table 7 shows the resilience criterion.

Table 6: The node operation criterion

Node operation	Detail	Security score (20/20)	Conclusion of security strength
Node addition	Key establishment and distribution for new node	10	Normal
Node revocation	Key erasure and update after node revocation	10	Normal
Node replacement	Both	20	Very strong

Table 7: The resilience criterion

Resilience after node compromise(nodes in danger)	Detail	Security score (20/20)	Conclusion of security strength
Whole network	Whole network is in danger	5	Weak
Parts of nodes	Parts of nodes	10	Normal
None	The keys can be updated safely	20	Very strong

3.5. Criteria for Efficiency and Operation

Besides security, the assessment of a key management scheme includes efficiency and operation due to requirements of key management in wireless sensor networks.

The criteria of efficiency have been established and admitted by the public. The general efficiency includes energy cost and memory cost. Energy consumption of a scheme depends on computation and communication cost while memory cost is related to number of keys stored in each node. The efficiency of a scheme is easy to assess because there are many simulation tools to help researchers to calculate costs of different resources. For computation, there are two parts: key calculation, which is the energy for key generation, and cryptography which is the energy for encryption and decryption of keying materials. Communication cost is the energy consumes for transportation of message exchange to generate and distribute keys. Conventionally, communication cost much more energy than computation [51]. On the other hand, memory cost is the space needed for key storage. The more keys a node holds, the more memory is required. And the required space increases when the key size grows.

Wireless sensor networks require flexibility, scalability and accessibility [52] which can be denoted as operation requirement. Key management schemes should be assessed in these too. Flexibility mostly depends on whether the scheme supports

node operation or not. In another words, the scheme with flexibility should provide node addition, revocation and replacement. Scalability is the scheme should support large scale of network, usually hierarchical structure is better than flat structure. Accessibility is each node is able to communication with each other. In key management, this feature is more depends on keying model. For single keying model schemes, network key is shared by all nodes, thus it has very good accessibility. In a scheme which employs group key, if it does not have any group shares no common node with other groups, the scheme has good accessibility too. In pairwise keying model scheme, if all nodes share pairwise key with each other, the accessibility is good. But if the pairwise key is partly employed in the network without other types of keys, the accessibility is critical. The key establishment approach also may cause critical accessibility too. For example, in Eschenaue scheme, people doubt its accessibility because not any pair of nodes is able to establish a shared key. Therefore, in mixed model schemes, only group key and pairwise key mixed schemes have to consider the accessibility carefully since both keying model may have accessible problem.

3.6. The Security Assessment for Panja and LEAP

Panja and LEAP are two popular key management schemes in current research. The two schemes are analyzed based on security criteria in this section. Panja only supports group key and basic encryption. It mentions both specific and regular update. The node operations are supported in this scheme. The resilience of Panja is not good. On the other hand, LEAP supports all keying models, the distribution of keys are securely carried out by different keys. The resilience against node compromise is good as all keys can update securely. But LEAP does not mention regular key update and node addition is unavailable. Thus LEAP has much higher score than Panja, its security is stronger than Panja. Table 8 shows the security assessment for the two schemes.

Table 8: The security assessment of LEAP and Panja

Security criteria	LEAP	Panja
Keying model (20)	All (20)	Group key only (10)
Key distribution (20)	Encryption by different key (20)	Basic encryption (5)
Key update (20)	Specific update (10)	Specific and regular update (20)
Node operation (20)	No addition (10)	All (20)
Resilience (20)	Security update (20)	Not good (10)
Total (100)	80	65

3.7. Summary

In this chapter, criteria for key management scheme assessment are proposed, which considers key management requirements of security, efficiency and operation. The preliminary criteria of security are proposed based on process of key management includes keying model, key distribution, key update, actions in node operation and resilience against node compromise. The current efficiency criteria are described. Memory cost and energy consumption of key calculation, cryptography and communication are the main efficiency measurement. The assessments of operation requirement in flexibility, scalability and accessibility are shown as well. In the next chapter, the structure of a wireless sensor networks is introduced and assistant node which is to improve security of a cluster after cluster head compromise is presented.

4. Cluster Based Wireless Sensor Networks and Key Management

Cluster based hierarchical wireless sensor network is a big category of wireless sensor networks with many potential applications [1]. In these networks, cluster heads not only aggregate application data, but also take management role which covers the security operations including the key management. Most existing key management schemes for hierarchical wireless sensor networks are based on the assumption that cluster heads are safe and reliable. However, in real world, cluster headers like normal nodes, can also be compromised. In this chapter, a modification on the cluster based hierarchical wireless sensor networks structure is provided by introducing a new management node in each cluster. This new management node is called assistant node. The introduction of assistant nodes is to handle the scenario when cluster heads are compromised.

4.1. Network Architecture

A sensor network typically consists of a large number of sensor nodes densely deployed in a region of interest, and one or more data sinks or base stations that are located close to or inside the sensing region. The base station sends queries or commands to the sensor nodes in the sensing region while the sensor nodes cooperate to accomplish the required task and send the sensed data to the base station. Meanwhile, the base station also serves as a gateway to outside networks, for example, the Internet. Base station collects data from the sensor nodes, performs simple processing on the collected data, and then sends relevant information (or the processed data) via the Internet to the users who requested it or use the information.

In wireless sensor networks, each sensor node can use single-hop long-distance transmission to send data to the base station. But long-distance transmission is expensive in energy consumption. In a sensor network, the energy cost on communication is much higher than it on computation. For example, the energy consumed for transferring one bit of data to a receiver at 100 m away is equal to that needed to execute 3,000 instructions [51]. The ratio of energy consumption for communicating 1 bit over the wireless medium to that for processing the same bit could be in the range of 1,000 – 10,000 [20, 53]. In addition, transmission energy cost dominates the total communication cost. The transmission energy cost increases exponentially with the growing transmission distance. Thus decreases of transmission distance and amount of communication are able to help reduce energy cost and prolong the lifetime of a network.

Therefore, the multi-hop short-distance transmission is a reasonable approach. In

most wireless sensor networks, nodes are densely deployed. Their neighbor nodes are close to each other and this provides possibility to use short-distance transmission. In this transmission, a sensor node transmits its data to the base station through one or more intermediate nodes. This reduces energy cost compares with data transmission from the node directly to the base station. The architecture of a multi-hop network can be classified as flat and hierarchical network [2, 54].

In a flat wireless sensor network, each node plays the same role to accomplish the required task. In order to collect required data, the base station sends a command to all nodes in the sensing region and the nodes that have the matching data to the command response the base station. Each node transmits data through multi-hop to the base station and uses other nodes as routers. Figure 7 shows a flat network.

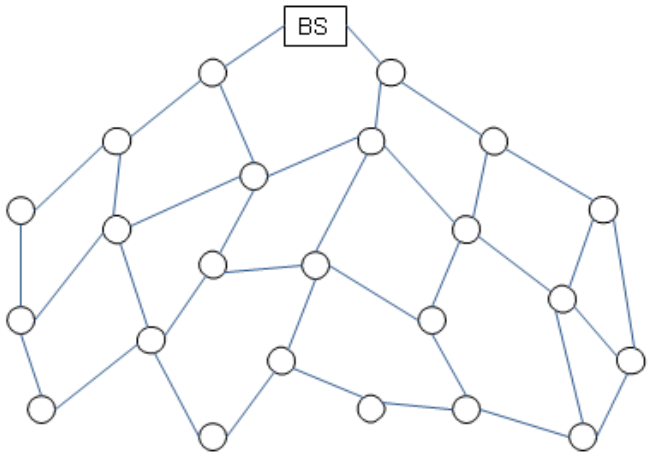

Figure 7: A flat wireless sensor network

In a hierarchical network, all sensor nodes are organized into clusters. Each cluster consists of cluster members and a cluster head. Cluster members send their data to the cluster head while cluster heads collect and aggregate data from member nodes, and then transmit data to the base station. This approach is able to reduce energy cost on communication and balance traffic load. Furthermore, data aggregation at cluster heads can help to reduce the amount of transmission as well as improve the efficiency of the network [2]. Figure 8 shows a hierarchical network. In a heterogeneous network [55] which consists of sensor nodes with different capabilities, cluster heads can be the nodes with higher resources in communication and computation. The capability of a node decides whether it is a normal member node or a cluster head. The deployment of this type of network should consider the locations of cluster heads and normal nodes. However in a homogeneous network which consists of nodes with the same capability, cluster head selection can be complex. In order to organize sensor nodes into cluster in a reasonable way to achieve efficiency, researchers have proposed many clustering algorithms.

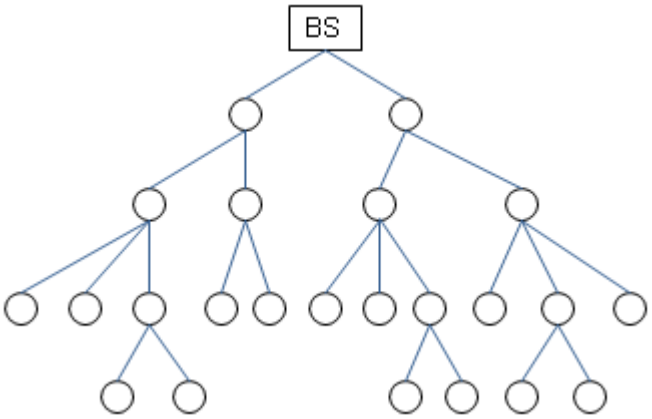

Figure 8: A hierarchical wireless sensor network

4.2. Node Clustering

As it is introduced in the last section, node clustering in wireless sensor networks has many advantages including less energy cost due to data aggregation, less routing information in normal nodes, more sleeping time for normal nodes and convenience in network management which is able to provide better scalability. However in a real wireless ad hoc environment, clustering algorithms face limitations such as unreliable and limited communication channel, and network topology changing. The clustering algorithms have to overcome these limitations.

LEACH (low-energy adaptive clustering hierarchy) [56] is the earliest node clustering algorithm in wireless sensor networks. Many other algorithms are proposed based on it such as TEEN (threshold sensitive energy efficient sensor network protocol) [57], HEED (hybrid energy-efficient distributed clustering) [58] and etc. There are many other algorithms different from LEACH such as ACE (algorithm for cluster establishment) [59] and LSCP (lightweight sensing and communication protocol) [60] .

The main steps of all these node clustering algorithms include: (1) cluster head selection; (2) cluster forming; (3) routing of nodes. In a node clustering algorithm, firstly cluster heads are chosen from the network. Cluster head selection is very important because appropriate cluster head selection can reduce the rate of re-clustering [61]. Then a cluster can be constituted by the cluster head and normal nodes affiliated to it. Nodes at the edge of a cluster are gateway nodes, which is able to communicate with other clusters. After the setup of clustering, the configuration of it can be changed if any node moves or network topology changes.

Cluster head selection is the basic of node clustering and also the first step of a clustering algorithm. In some heterogeneous network [62] clustering algorithms, cluster heads are pre-specified and deployed where they have unlimited energy

resources. But in homogeneous node clustering algorithms, cluster heads are nodes with limited resources and nodes have the same capability. In order to prolong the lifetime of a network, cluster heads are re-selected periodically. The selection algorithm decides the number and location of cluster heads in a network which affects number, size and structure of clusters, and therefore decides the lifetime of the network. The current cluster head selection algorithms are proposed depending on four features: the remained energy, the distance between the node to the base station, the deployment of the node (includes the coverage and connectivity to other nodes) and communication cost inside the cluster.

Cluster heads are the most important nodes in their clusters. Cluster head related operation can be summarized in the following [63].

- **Efficient organization:** In a large scale hierarchical wireless sensor network, it is very resource consuming for base station to communicate directly with all normal nodes in the network. However, through cluster heads the network can efficiently organize all normal nodes and base station can efficiently communicate with normal nodes.
- **Data aggregation:** Neighboring nodes can easily collect highly correlated data, in order to reduce redundant data transmission in the network, data aggregation is essential. Cluster heads usually process data aggregation as all data from normal member nodes have to pass through heads.
- **Security management:** Cluster heads usually generate and distribute security keys. For example, in Panja scheme, cluster heads generate and distribute group key of the cluster. Once cluster head is compromised, the whole cluster will be in danger because all keys in cluster head are disclosed and there is no node in the cluster can update the key.

As a cluster head plays such important roles, the compromise of a cluster head is fatal in the cluster and it affects all the nodes in the cluster. In order to combat the security problem caused by cluster head compromise, an assistant node based management is proposed.

4.3. Assistant Node in Hierarchical Wireless Sensor Networks

In a cluster, besides the cluster head, another management node, named assistant node (AS) is arranged. An assistant node is randomly chosen by the cluster head at cluster setup phase. It should have pairwise keys with each member nodes and the cluster head. The pairwise key establishment will be covered in the next chapter. All member nodes in the cluster are candidates to be an assistant node. Whenever a

cluster head is compromised, replaced or removed, the assistant node will kick in and will take the management role in the cluster. Figure 9 shows assistant nodes in a three-layer wireless sensor network.

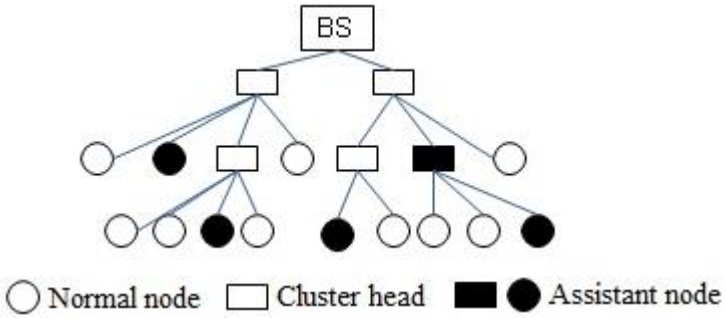

Figure 9: Assistant nodes in a three- layer hierarchical wireless sensor network

An assistant node has IDs of all nodes in the same cluster as well as the information of gateway nodes. The gateway nodes enable the assistant node to find routes to the upper level head above its own cluster head in the hierarchical structure.

In heterogeneous networks, there is no need for re-clustering as cluster heads have enough resources and they are selected before deployment. After cluster setup, cluster head randomly chooses a normal node as the assistant node. The cluster head sends IDs of all the other nodes and information of gateway nodes to the assistant node.

The assistant node finds a route to the head above its own cluster head at clustering stage in 6 steps:

1) Initialization
   - Set Searching Depth to 0;
   - Create two route request packets with the following parameters
     i. Cluster head ID (source CH ID)
     ii. Routing Path : Assistant Node ID
     iii. Searching Depth
2) Assistant node passes the requests packet to a gateway node never encountered before.
3) The gateway node adds its own ID to the routing path and sends the request to a gateway node in a neighboring cluster.
4) Then this neighboring gateway node adds its own ID to the Routing Path and passes the request packet to its cluster head.
5) Each time a cluster head receives a request, it will increase Searching Depth by 1. Then it will check if the source CH ID is its member. If it is, then the route is found. Otherwise it will check if Search Depth is larger than 3. If it is then go back to step 1. Otherwise it adds its ID to the Routing Path and passes the request to its own head, then goes back to step5.
6) End.

Figure 10 (a) shows the routing when assistant node is a gateway node. In (a), route 1 is able to reach the head of cluster head, so it should be the route return to the node A. However in Figure 10 (b) route 1 is failed because the cluster of node B is at the fringe of the higher-layer cluster and its neighbor belongs to another head of cluster heads. Thus route 1 is a wrong route and route 2 would be setup after the failure of route 1.

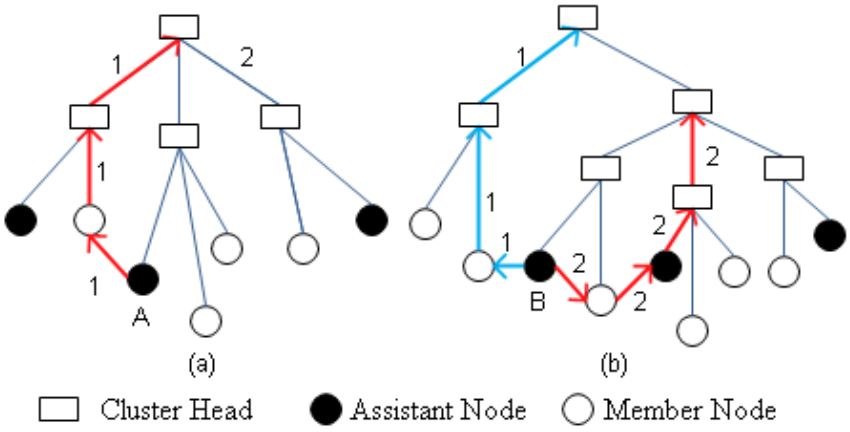

Figure 10: Routing from assistant nodes to head of cluster heads

In homogeneous network, since all sensor nodes have the same capability, the network has to re-cluster periodically to distribute energy overhead evenly on all nodes [64, 65]. If an assistant node always tries to find a route to the head of its cluster head after each re-clustering, the energy consumption of routing for assistant nodes would be too much. Thus in homogeneous network, assistant nodes remember information of the gateway nodes in the same cluster. It will find routes to the head of cluster head when the cluster head is compromised or damaged. The routing approach is similar with the one in a heterogeneous network, but the response from the head of cluster head should include the information of whether the member nodes join the other cluster or wait for a new cluster head to replace the old one.

In this assistant node management, an assistant node is able to connect other member nodes in the cluster securely. Once a compromised cluster head *i* is detected, the assistant node in the cluster is able to communicate with the head *j* above its own cluster head *i* by pairwise key. This head *j* is the leader of *i* and should be able to manage *i* and other cluster heads in the same layer with *i*. If there is a newly adding node *k* to replace the compromised node *i*, the new node *k* would receive a temporary group key *GK* from cluster head *j* after join its cluster. Head j also sends *GK* and command to join the cluster of node *k* to the assistant node. Then the assistant node distributes the *GK* and command in the cluster secured by pairwise key with each member node. Member nodes can join the cluster of head *k* after authenticated by *GK*. If there is no new node to replace the compromised head *i*, head *j* sends *GK* to *i*'s neighboring cluster heads. The assistant node receives *GK* and command to join

neighbor clusters, and then distributes them in the cluster. Member nodes can join the nearest cluster. This approach helps member nodes to get out of control from the compromised cluster head.

4.4. Summary

In this chapter, the architecture of wireless sensor networks is first introduced. A network usually consists of a base station and many sensor nodes which can be structured as a flat network or a hierarchical network. Then general node clustering algorithms in hierarchical networks are presented. The main steps of algorithms are almost the same, which include cluster head selection, cluster forming and routing. After that, the assistant node management is proposed and its evaluation is presented too. In the next chapter, a newly proposed key management scheme will be proposed and assistant node management is applied in the new scheme.

5. The Secure Efficient Hierarchical Key Management Scheme

5.1. Introduction

The existing key management schemes are analyzed in the previous chapters. In this chapter, a new secure efficient key management scheme for hierarchical wireless sensor networks is proposed based on LEAP [34] and Panja [39]. LEAP is a mix model scheme while Panja is a single model scheme in hierarchical networks. The proposed scheme has kept the good features of both schemes. LEAP is a better scheme in supporting services to secure different types of communication. In wireless sensor networks there are many communication types, such as base station broadcast messages to all network nodes, cluster multicast messages to all cluster members and unicasts messages from member node to cluster head. LEAP provides four types of keys to secure different communication types. One problem of LEAP is that it is not flexible because it assumes a static network and it is not efficient. Panja is the key management scheme supporting establishment of group key in hierarchical structure of wireless sensor networks. This structure helps to reduce message transmission due to data fusion at cluster heads. So it is more efficient than flat structure. Panja is more efficient but its security is weak. It only supports one type of key, and thus if the key is compromised, the whole network will be in danger. A key management scheme needs to satisfy security, efficiency and operation requirements and it is hard to find one scheme to achieve all of them at the same time. Based on recent researches and the key management assessment criteria proposed in chapter 3 which have shown that key generation, distribution, update and node operations are significant to a key management scheme. In this chapter, LEAP and Panja are analyzed in detail. Then a new scheme is proposed. Key generation, distribution, update and node operations of the proposed scheme are clearly described.

5.2. Analysis of Existing Schemes

In this section, two schemes, ie Panja and LEAP are analyzed and their problems are outlined. For each scheme, the operation of key generation, distribution and update are described. The advantages and weaknesses of the schemes are also analyzed.

5.2.1. Problem of Panja

Panja is a dynamic single model key management scheme. It only includes one type of key: group key. This key is shared by a group of nodes which is a cluster. Each node in the cluster contributes a partial number to generate the group key. Leaf nodes generate random numbers as their partial numbers. Then the parent of the leaf nodes computes its partial number using a function. In Figure 11, the leaf nodes

are $M_1, M_2, \ldots, M_9$. To start, M_1 computes the partial key $f(S_1)$ and broadcasts it. The parent node M_{10} gets the partial keys from M_1 and other children. Here, f is a generator algorithm using g from the multiplicative group ZP* (i.e. the set {1, 2… $p-1$}, p is the prime) and S_1 is a randomly chosen secret number for member M_1.

$$f(k) = g^k \bmod p \qquad 5.1$$

Likewise, member M_2 computes $f(S_2)$and broadcasts it, and the parent M_{10} gets the partial keys. In this way, member M_{10} receives $f(S_1)$, $f(S_2)$, $f(S_3)$, and raises the power by S_{10} to get the intermediate key IK_1. Here, $f(S_{10})$ is the partial key contribution of M_{10}. IK_1 is generated d by $f(S_1)$, $f(S_2)$, $f(S_3)$and $f(S_{10})$.

$$IK_1 = f\big(f(s_1), f(s_2), f(s_3), f(s_{10})\big) = g^{f(s_1)f(s_2)f(s_3)f(s_{10})} \bmod p \qquad 5.2$$

The intermediate keys in M_{10}, M_{11}, and M_{12} are IK_1, IK_2 and IK_3, they are generated by the partial keys of member nodes. The intermediate keys are encrypted using a one-time symmetric key at the first time. The following ones are encrypted by the last group keys. The cluster head generates the group key K, using IK_1, IK_2 and IK_3 and its contribution $f(S_{13})$.

$$K = g^{IK_1 IK_2 IK_3 f(S_{13})} \bmod p \qquad 5.3$$

Figure 11 shows the structure of Panja.

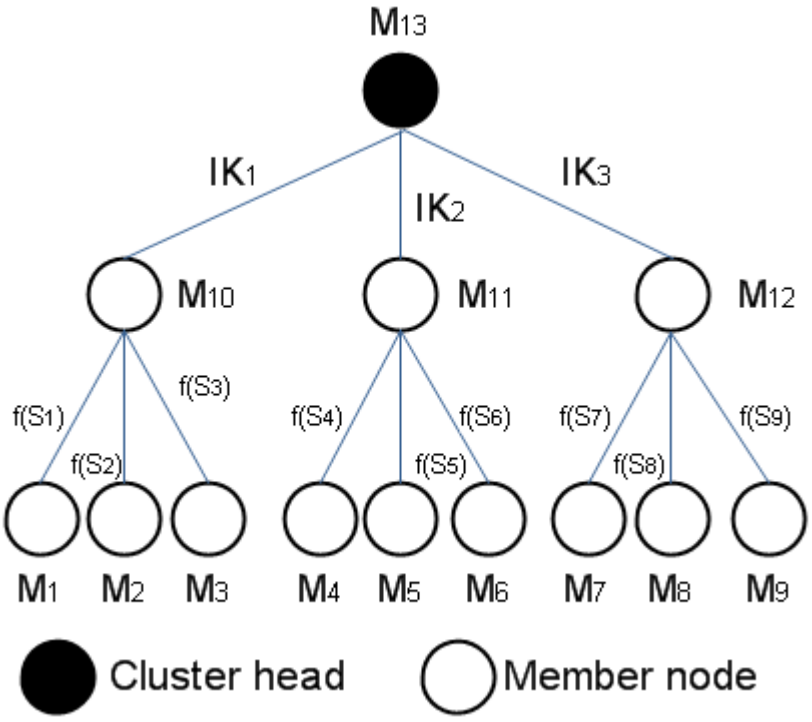

Figure 11: Panja scheme

Panja also includes group key updating. The approach to keep the key fresh is by re-keying the group at fixed intervals; in this approach the partial keys and the group key will be computed again. The group key establishment uses the described algorithm

above which is secure but not efficient as it costs too much energy on communication. Firstly, BS broadcasts the key update request to all cluster heads, and then cluster heads broadcast the message to all member nodes. Secondly, member nodes send partial numbers to cluster heads. Then cluster heads calculate group keys. Next cluster heads broadcast the group key in the cluster and send their seeds to BS to generate group key in higher level. Finally BS generates group key and broadcast it in the group. All broadcast encrypted by old group key. In this approach, each cluster head has to broadcast key update request and at last broadcast the new group key; each node has to send its partial number to cluster head. This costs too much energy on communication.

Besides costly key update, another main problem of Panja is weak security. The whole cluster can be in danger if any node is compromised because all nodes in the cluster just have one key. Even the cluster head try to update the key, the new key would be disclosed because new key distribution encrypted by old key.

5.2.2. Problem of LEAP

As it is introduced in section 2.5.4, LEAP includes four types of keys: individual key, group key, cluster key and pairwise key. Different keys are used for encryption of different types of packets.

- **Individual key:** This is a unique key shared by base station and each node. It is pre-distributed into each node before its deployment. The individual key shared by a node *u* and base station can be calculated as:

$$K_u^m = f_{K^m}(ID_u) \quad 5.4$$

  f is a pseudo-random function and K^m is the master key only known by base station.

- **Pairwise key:** This is a key shared by each node with its each immediate neighbor. The establishment of pairwise key includes four steps: key pre-distribution, neighbor discovery, pairwise key establishment and key erasure. In the first step, node u is pre-loaded with K_i which helps it to generate master key K_u. K_i is an initial key generated by controller.

$$K_u = f_{K_i}(ID_u) \quad 5.5$$

  The next step is neighbor discovery, node *u* set timer at time t_{min}. Then node *u* broadcasts a HELLO message including its *ID*. The neighbor node *v* receives the message and responds to node u with an acknowledgement (ACK) including its *ID*. This ACK is authenticated by master key K_v which is derived from K_i. Node u verifies it by K_v which is generated by received *ID* of *v* and K_i. The third step is pairwise key establishment. The pairwise key shared by node *u* and *v* is:

$$K_{uv} = f_{K_v}(ID_u) \quad 5.6$$

Node u and v calculate the key by this function. In the last step, the timer expires after t_{min}. Node u erase K_i and all the master keys of its neighbors.

- **Cluster key:** This key is shared by a node and all its immediate neighbors. Cluster key establishment after the pairwise key establishment. This key is a random number generated by node u. Then node u transports the key to all its neighbors encrypted by their pairwise key.
- **Group key:** This key is shared by base station and all nodes in the network. LEAP uses a cluster based scheme to distribute the key.

The main problems of LEAP include lack of flexibility due to key erasure. In this scheme, each node has to use master key which is derived from an initial key K_i to join the network. But all nodes erase master key and the initial key K_i after a specific period of time which means any node can only join the network in this period and no node is able to join after the period. Moreover, if initial key K_i is disclosed, the adversary will be able to establish pairwise key with any node in the network. In addition, on key update, LEAP scheme just generates a pseudo-random number as a new cluster key which is not secure because cluster key is frequently used. The security of the key highly relies on pseudo-random number generation algorithm. Current research on this area shows that the numbers generated by any algorithms are repeating and can be broken by potential adversary. An adversary only needs to compromise one node to get the pseudo-random number generation algorithm.

5.3. SEHKM Scheme

Based on LEAP and Panja which described in chapter 3, a secure efficient hierarchical key management scheme for wireless sensor networks (SEHKM) is proposed. This scheme combines the advantages of LEAP and Panja, therefore it is a hierarchical scheme which supports three types of keys and defines key generation, transportation, update and node operations.

The assumptions of the wireless sensor network scenarios that the proposed scheme can be used in are described here. Firstly, it is a dynamic hierarchical sensor network. There are node addition, revocation and replacement after network initialization. Secondly, the base station of network is a device with enough resources and always under the control of controller safely. The initialization phase of network is safe too. Then each node has a unique pre-loaded identity (*ID*), and the *ID*s which belong to removed and replaced nodes cannot be used again. After network clustering, a cluster head is able to get all *ID*s of member nodes. Next is all nodes beside base station can be compromised and the information in which will be exposed to the adversary.

Here is a list of notations which appear in the following discussion.

- *BS* is the base station.

- CH_i is a cluster head i.
- N_u is a participant node u.
- C_i is a cluster which cluster head is node i
- NK is the network key.
- IN is an initial number.
- GK_n is the group key of the cluster which cluster head is n.
- $PK_{x,y}$ is the pairwise key of node x and y.
- $\mathrm{DPK}_{u,i}$ is the disposable pairwise key shared by node u and i
- $f_m(x)$ is a pseudo-random function
- $f(k_1, k_2)$ is a function, in which p is a prime number and α is its primitive root.
- $(s)_k$ means message s is encrypted by key k.
- N^+ is positive integer set

SEHKM scheme is presented in a two-layer hierarchical wireless sensor network.

5.3.1. Keys Types

Three types of keys in SEHKM are supported. They are network key, group key and pairwise key.

- **Network key.** Network key is the key shared by the whole network. It helps with the initialization of a network. Besides, it is used for authentication and encryption of broadcast messages. Network key is pre-distributed in each node and employed for authentication. It should be renewed each time of key update and after any node operation.
- **Group key.** Group key is shared by a group of nodes and each cluster is a group. In two-level hierarchical wireless sensor networks, a cluster head and all normal nodes in the same cluster is a group and share one key. All cluster heads and base station is a group and share one key too. Group key encrypts multicast message in the group. It is generated by group leader and should be renewed each time of key update and after any node operation in the cluster.
- **Pairwise key.** Pairwise key is a key shared by specific pair of nodes which protects unicast messages. In SEHKM, not any pair of nodes has a pairwise key. Base station and each node shared one key, cluster head and each cluster member shared one key and assistant node and each cluster member shared one key. The pairwise keys shared by group leader and its member nodes are generated by Diffie-Hellman algorithm and stored in every participant. Moreover the pairwise key shared by base station and each node is pre-loaded in the node and generated in base station when it is needed. It is the same as the key shared by assistant node and member nodes in the same cluster.

5.3.2. Key Pre-distribution

To satisfy key management requirements on security, pre-distribution of keys are essential to keep secrecy in initialization of a network. In SEHKM scheme, all nodes have to pre-distribute two keying materials: the network key and an initial number *IN*.

During initialization phase, the pre-load *IN* and network key help to setup the network. After initialization of the network and distribution of all keys, network starts requiring application. In the case of node compromise, damage or flat battery, node addition is required. A new node is pre-loaded with current network key and *IN* which can help the node to join the network. Without authorization using these two keying materials, any joining requests from nodes should be rejected. In order to keep secrecy, *IN* and the network key will be updated by BS after a period of time.

Network key is updated periodically and shared by all nodes in the network while *IN* is pre-distributed in each node during initialization phase and erased from every node after all keys are established. Then *IN* is updated periodically and only known by base station and newly adding nodes. These nodes have to erase *IN* after join the network.

5.3.3. Initialization

In SEHKM, the pre-distributed network key helps with node clustering of the network and disposable pairwise key establishment by *IN* follows.

5.3.3.1. Disposable Pairwise Key Establishment

In the network clustering, each member node u joins the cluster of the head node i and generates the disposable pairwise key by *IN* and its *ID* using function f.

$$DPK_{u,i} = f_{IN}(ID_u) \tag{5.7}$$

As a cluster head knows all *ID*s of its member nodes, it is able to calculate the disposable pairwise key with each member node. After network clustering and generation of disposable pairwise key with head node, each node erases *IN* and only remains disposable pairwise key.

5.3.3.2. The First Group Key Establishment

Group key is shared by a group of nodes. Base station and all cluster heads constitute a higher level group, each cluster head and its member nodes in the same cluster form a group. In another word, a cluster is a group and cluster head is the group leader. The group leader generates group key and broadcasts it in the group. The first group key is a random number generated by group leader and its transportation is encrypted by disposable pairwise key. The algorithm that generates the following group keys is presented in section 5.3.4.

5.3.3.3. Network Key Establishment

After group key employment, base station generates a random number as the new network key. Base station encrypts the key by group key and broadcasts the key to all cluster heads, then each cluster head broadcasts the key encrypted by its group key to its own member nodes.

The pseudo-code of the above process can be put as following:

Disposable pairwise key establishment:

For each cluster head i, $u \in C_i$, $i \in C_{BS}$
{ $DPK_{u,i} = f_{IN}(ID_u)$
For each node u,
{ $DPK_{u,i} = f_{IN}(ID_u)$ }
}

Group key establishment:

For each cluster head i, i $\in C_{BS}$
{ $GK_i = KG(l)$ 'Generation of group key'
For cluster head sends GK to each node u, u $\in C_i$
{ $CH_i \rightarrow N_u : (GK_i)_{DPK_{u,i}}$ }
}

Network key update:

$NK = KG(l)$ 'Generation of network key in BS'
$BS \rightarrow * : (NK)_{GK_{BS}}$ 'BS broadcasts NK to all cluster head'
For each cluster head i broadcasts NK to all node, $i \in C_{BS}$
{ $CH_i \rightarrow * : (NK)_{GK_i}$ }

5.3.3.4. Pairwise Key Establishment

There are three types of pairwise keys in the scheme which distinguished by their generation and storage. The first type is group associated pairwise keys which shared by group leaders with their member nodes and the pairwise keys shared by BS with cluster heads. The second type of pairwise keys is assistant node associated. This pairwise key is shared by assistant node with member nodes in the same cluster and the key shared by assistant node with BS. The last type is disposable pairwise key and it will be erased after the establishment of all keys.

- **Group associated pairwise key establishment:**

The generation of group associated pairwise key uses Diffie-Hellman algorithm. For example, in a group, the leader i generates a random seed g_{1i}, then calculates g'_{1i} as:

$$g'_{1i} = \alpha^{g_{1i}} \bmod p \qquad 5.8$$

The group leader broadcasts g'_{1i} encrypted by group key in the group. Then a member node *u* generates g_{2u} and sends g'_{2u} to group leader *i*. g'_{2u} is encrypted by the disposable pairwise key. Therefore group leader gets g'_{2u}. Then the pairwise key between member node *u* and group leader *i* is:

$$PK_{u,i} = f(g_{1i}, g_{2u}) \quad 5.9$$

And the function f is:

$$f(k_1, k_2) = \alpha^{k_1 k_2} \bmod p \quad 5.10$$

In this function, p is a prime number and α is the primitive root of p. $k_1, k_2 < p$. α and p are known by all nodes.

Cluster head *i* has g_{1i} and g'_{2u}, the *PK* is :

$$PK_{u,i} = \alpha^{g_{1i} g_{2u}} \bmod p = (\alpha^{g_{2u}} \bmod p)^{g_{1i}} \bmod p = {g'_{2u}}^{g_{1i}} \bmod p \quad 5.11$$

Node u has g_{2u} and g'_{1i}, the *PK* is:

$$PK_{u,i} = \alpha^{g_{1i} g_{2u}} \bmod p = (\alpha^{g_{1i}} \bmod p)^{g_{2u}} \bmod p = {g'_{1i}}^{g_{2u}} \bmod p \quad 5.12$$

After generation of pairwise key, it is kept by both group leader and member node. All cluster heads will erase the disposable pairwise key shared with each member node after a fixed period of time.

The pseudo-code of process of establishment of pairwise key shared between group leader and its member node can be shown as:

For each cluster head *i*, $i \in C_{BS}$
{ $CH_i \rightarrow * : (g'_{1i})_{GK_i}$
For each node u, $u \in C_i$
{ $N_u \rightarrow CH_i : (g'_{2u})_{DPK_{u,i}}$
$PK_{u,i} = f(g_{1i}, g_{2u}) = {g'_{1i}}^{g_{2u}} \bmod p$ '*u* gets *PK* with *i*'}
For cluster head *i* calculates *PK* shared with each node u
{ $PK_{u,i} = f(g_{1i}, g_{2u}) = {g'_{2u}}^{g_{1i}} \bmod p$ '*i* gets *PK* with *u*'}
}

- **Assistant node associated pairwise key establishment:**

An assistant node is selected beside the cluster head in each cluster. The assistant node associated pairwise key is generated by a random function. $PK_{u,v}$ is pairwise key shared by node *u* and *v*. *m* is a number only known by generator node *v*. The function is:

$$PK_{u,v} = f_m(ID_u) \quad 5.13$$

Then generator node *v* sends $PK_{u,v}$ to node *u*. Node *u* stores $PK_{u,v}$ while node *v* only stores node *u*'s *ID*. It will generate $PK_{u,v}$ if it is needed.

For a key shared by the assistant node and a member node, the assistant node plays the role of generator. However BS is the generator for the pairwise key between BS and an assistant node. After establishments of all pairwise keys, *IN* and disposable pairwise keys will be erased after a period of time.

The pseudo-code of establishment of assistant node associated pairwise key is presented in the following:

Pairwise key shared by assistant node and normal node:

For each assistant node a, $a \in C_i$, $i \in C_{BS}$

{ $CH_i \rightarrow N_a : (ID_{u_1}, ID_{u_2}, ID_{u_3}, \dots, ID_{u_j})_{PK_{i,a}}$, $\{u_1, \dots, u_j, a, i\} = C_i$

For assistant node a gets pairwise key shared with each node j, $j \in C_i$

{ $PK_{a,j} = f_m(ID_j)$, $m \in N^+$ 'a gets pairwise key shared with j'

$DPK_{a,j} = f_{IN}(ID_j)$ 'a is able to calculate the disposable pairwise key'

}

For assistant node a sends PK to each node j

{ $N_a \rightarrow N_j : (PK_{a,j})_{DPK_{a,j}}$ }

}

PK shared by BS and an assistant node:

For each cluster head i, $i \in C_{BS}$

{ $CH_i \rightarrow BS : (ID_a)_{PK_{BS,i}}$, $a \in C_i$ }

For BS generates PK shared with each assistant node a

{ $PK_{BS,a} = f_m(ID_a)$, $m \in N$ 'BS gets each PK shared with assistant node'

$DPK_{BS,a} = f_{IN}(ID_a)$ 'BS is able to calculate the disposable pairwise key'

}

For BS sends PK to each cluster head i, $i \in C_{BS}$

{ $BS \rightarrow CH_i : ((PK_{BS,a})_{DPK_{BS,a}})_{PK_{BS,i}}$ }

For cluster head i, it sends PK to assistant node a, $a \in C_i$

{$CH_i \rightarrow N_a : ((PK_{BS,a})_{DPK_{BS,a}})_{PK_{i,a}}$}

After this step, all keys are established. *IN* and the disposable pairwise key will be erased. The initialization of keys has completed.

5.3.4. Key Update

After the initialization phase, the key update is the key management operation. Key update is carried out in regular basis or in one of the following circumstances, ie node addition, revocation and replacement. A list of notations is defined below before the different types of key update are presented.

- BS : base station.
- CH : cluster head.
- MN : member node.
- NK : network key
- GK: group key.
- PK : pairwise key.
- DPK: disposable pairwise key.
- IK : temporary key.
- SRN: small random numbers

5.3.4.1. Regular Key Update

Regular key update is significant to keep secrecy of keys. However each sensor node has limited energy supply on a battery which is difficult or even impossible to be replaced or recharged in many applications. So the resources cost of key update has to be reasonable in efficiency consideration. In this section, a method which helps to reduce energy cost in key update is proposed.

For the purpose of efficiency, in SEHKM, only the network key and group keys will be updated periodically. Network key is generated by the base station while group key is established by collaborating of all nodes in the cluster. The mechanism of the group key generation in the upper layer is the same as in the cluster level. Therefore the following discussion on the proposed key update algorithm focuses on the group key update on the cluster level. The main feature of the key update is that member nodes send small random numbers (SRNs) along with normal traffic data together to reduce transmission energy. In current cryptography algorithm, some of them using block cipher technique which is treats a block of plaintext as a whole to be encrypted [66]. If the message is not long enough as the block, 0 is used to fill the empty parts. So these empty parts can be used to carry the SRNs.

The group key update includes four steps: (1) member nodes send SRNs to the cluster head; (2) the cluster head stores and transforms SRNs; (3) the group key generation; (4) the group key distribution.

1) In the normal network operation, a member node attaches an r-byte SRN with its normal application data regularly. Not all packets have a SRN attached. The frequency of SRN attachment at each node is fixed.

2) A cluster head stores these SRNs collected and assembles a number N_1 of K bytes. K is the key size where $K>r$. If only $r-i$ bytes of last SRN are used, then the remaining i bytes will be used for assembling N_2. When $N_2$is assembled, the function f is used to generate a new number and N_1 is updated with this new number. Only the very first N_1 is assembled from the SRNs collected.

The following N_1 is from the recursive function f 5.10 using the last N_1 and N_2:

$$N_1 = f(N_1, N_2) \qquad 5.14$$

The following SRNs are used to assemble N_2 only. SRNs are from different nodes. The order of them is not important as long as they are not from one node only. If a node sends SRNs more than [*K/n*] times continually, the cluster head should erase [*K/2*] bytes and generates [*K/2*] bytes of random number to re-assemble N_1 or N_2. The process keeps going till the cluster head receives a command to update the group key. Then all the following SRNs are discarded.

Let us assume a cluster *C* has L nodes, $C = \{1, 2, 3 \ldots L\}$ and K=8, r=3. Figure 12, $n_1 \neq n_2$, $n_3 \neq n_4$, $n_i \in C$, $1 \leq i \leq L$, the initial N_1 assembling is shown as:

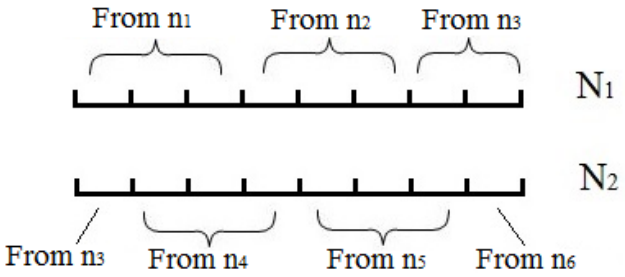

Figure 12: The generation of the initial N_1

After the initial N_1 is assembled, the following N_1 is generated by function f and N_2 is always assembled from SRNs.

3) The third step is group key generation. When the cluster head needs to update the group key, if the SRNs received are not enough to form a new N_2, then the cluster head pads N_2 with a generated random number. Figure 13 shows the last N_1 and N_2 before the key update. If the N_2 is empty, which means the last N_2 is just assembled and new N_1 is calculated, then the cluster head generates a K-byte number as N_2. In Figure 13, $x \in C$.

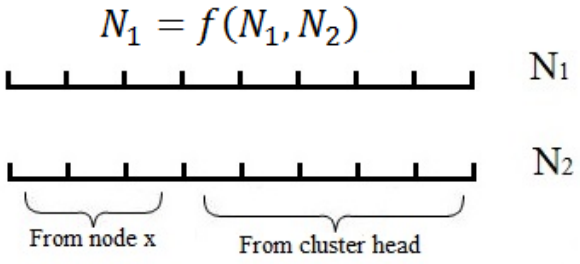

Figure 13: The generation of the last N_1 and N_2

The group key is generated by the function f 5.10 using the last N_1 and N_2.

$$GK = f(N_1, N_2) \quad 5.15$$

If a node compromise or damage is detected, then the N_1, N_2 generated from SRNs should be discarded as it is hard to figure out which parts of them are from the compromised nodes. Then the cluster head will use the new network key as N_1 and generates N_2 by itself to calculate a new group key.

4) The cluster head distributes the new group key and new network key together. Here it is assumed that the new network key has been generated in BS and distributed to the cluster heads. The key distribution can be used by previous group keys. However, if node compromise is detected, the distribution should be carried out securely by pairwise keys.

5.3.4.2. Node Addition

A new node pre-installs the current network key and *IN*. The disposable pairwise key can be calculated using function f as it is described in section 1.3.3.1. The new node has to be authenticated by network key and disposable pairwise key to join the network. Figure 14 shows the addition process of normal node.

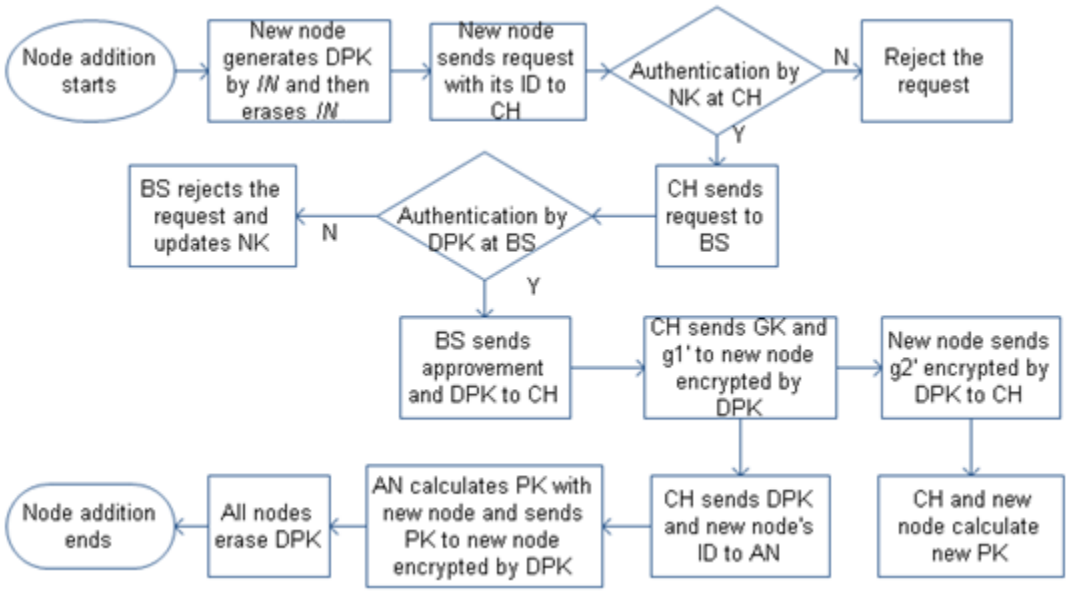

Figure 14: Node addition

The pseudo-code of adding node u can be shown as:

Node addition:

Pre-load *IN* and *NK* into node *u*

Node *u* generates disposable pairwise key and then erases *IN*

$DPK_{BS,u} = f_{IN}(ID_u)$

Node *u* sends join request to cluster head *i*, $i \in C_{BS}$

$N_u \rightarrow CH_i : join\ request, ID_u, (ID_u, (ID_u)_{DPK_{BS,u}})_{NK})$

If the message is verified by NK in cluster head *i* **then**

{ $CH_i \rightarrow BS : (ID_u, (ID_u)_{DPK_{BS,u}})_{PK_{BS,i}}$

If the message is verified by $DPK_{BS,u}$ in BS **then**

{ BS $\rightarrow CH_i : (approvement, DPK_{BS,u})_{PK_{BS,i}}$

Node *u* gets GK_i and sends to generate $PK_{u,i}$

$CH_i \rightarrow N_u : (GK_i,\ g'_{1i})_{DPK_{BS,u}}$

$PK_{u,i} = {g'_{1i}}^{g_{2u}} \bmod p$

$N_u \rightarrow CH_i : (g'_{2u})_{DPK_{BS,u}}$

Cluster head generates $PK_{u,i}$

$PK_{u,i} = {g'_{2u}}^{g_{1i}} \bmod p$

Assistant node *a* receives *ID* of *u*, generates $PK_{a,u}$ and sends it to u

$CH_i \rightarrow N_a : \left(ID_u,\ DPK_{BS,u}\right)_{PK_{i,a}}$

$PK_{a,u} = f_h(ID_u)$

$N_a \rightarrow N_u : (PK_{a,u})_{DPK_{BS,u}}$

All nodes erase $DPK_{BS,u}$

}

else if BS rejects the request and update *NK* **then**

BS $\rightarrow CH_i : (reject)_{PK_{BS,i}}$

else cluster rejects the request

}

5.3.4.3. Node Revocation

Node revocation includes two steps: the first step is to remove the node from network, the second step is to update all keys that the removed node has. During the time period after node removal and before key update, node addition is not allowed. The node revocation process is shown in Figure 15.

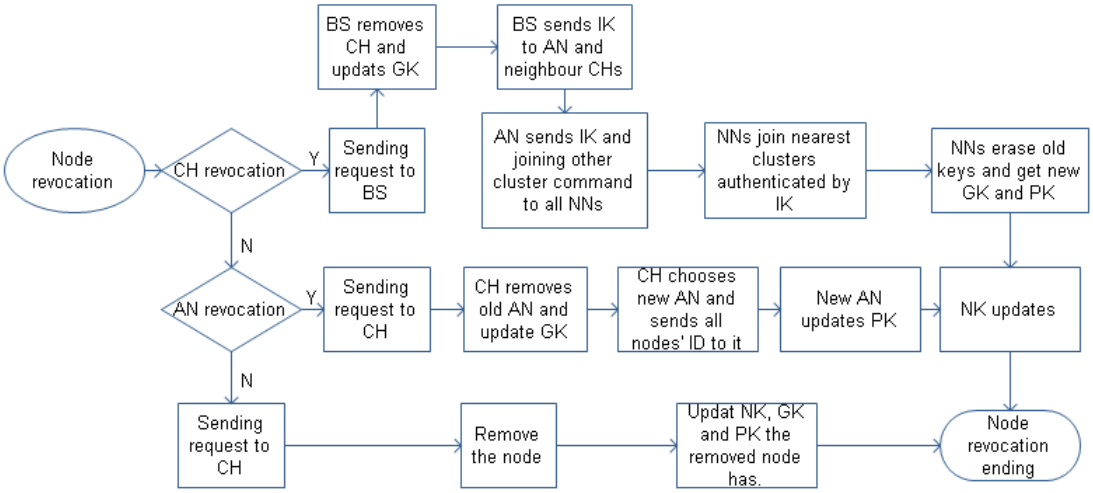

Figure 15: Node revocation

If the removed node is a cluster head, base station sends a temporary *IK* to assistant node and cluster heads around. Assistant node distributes the *IK* to all normal nodes and let them join the nearest cluster to them. Nodes may join different cluster and they use *IK* to authenticate. Then these nodes establish pairwise key with new head and assistant node. They also receive new group key. All the old keys are erased. At last, the network key is updated.

If an assistant node is removed, the head will updates the group key and distributes it by pairwise key with each node; then it picks up a new assistant node and gives its all *ID*s of other nodes; next the new assistant node generates pairwise keys shared with other normal nodes. The BS updates the network key after the cluster key is updated.

If it is a member node, the cluster head just updates group key and erases its pairwise key. Then the BS updates network key.

5.3.4.4. Node Replacement

Node replacement acts as node addition and then node revocation. The process is shown in Figure 16.

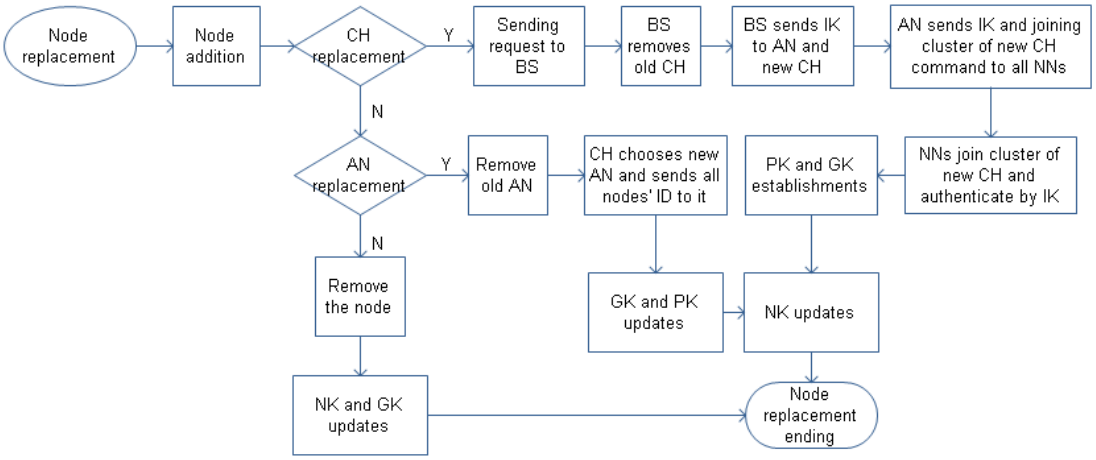

Figure 16: Node replacement

5.4. Summary

In this chapter, Panja and LEAP schemes are further analyzed. The drawbacks of the two schemes are identified. A new secure efficient hierarchical key management scheme (SEHKM) is proposed. It covers key type definition, key generation, key distribution, regular key update and key update in node addition, revocation and replacement. The proposed scheme is aimed to offer strong security and efficient performance. The evaluation of the proposed scheme will be presented in the next chapter which includes security assessment, efficiency assessment and operation assessment.

6. Evaluation of SEHKM

In this chapter the SEHKM scheme is evaluated. Firstly, the efficiency of SEHKM is analyzed and simulation results are shown. Then the security of the proposed scheme is assessed based on the criteria proposed in chapter 3. Lastly, the operation assessment of the scheme is presented.

6.1. Performance Analysis

In SEHKM, the first network key and group key are random numbers that generated by the base station and cluster head. Pairwise key establishment and distribution only carry out once. The energy cost on these processes is fixed and not expensive. However network key and group key are updated periodically, the energy cost on each time of key update should be reasonable to satisfy efficiency requirement. The network key is a pseudo-random number that generated by the base station and distributed cluster by cluster. The energy cost on network key update is reasonable. The group key update algorithm in SEHKM is similar with Panja scheme, but the proposed algorithm using the empty parts in block cipher technic to send keying materials which is much more efficient. Figure 17 shows the process of group key update in Panja and the proposed scheme.

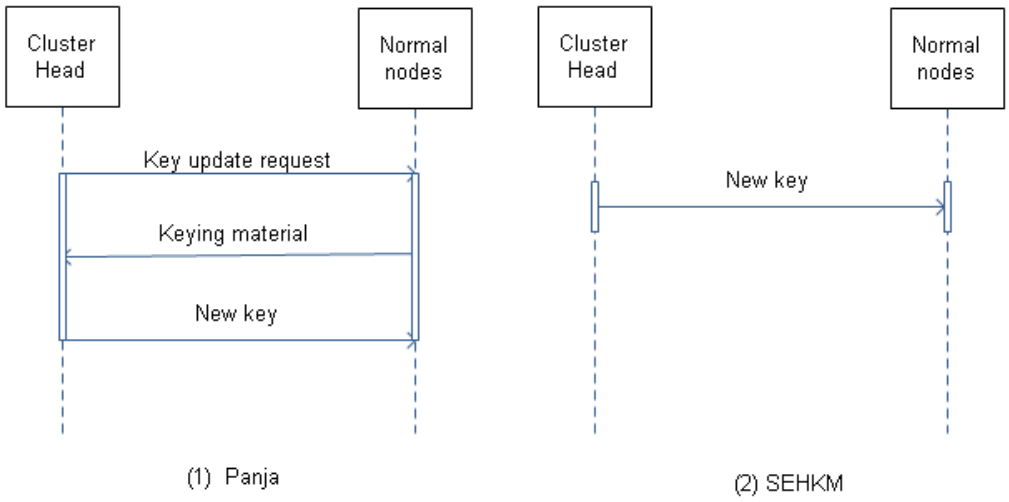

Figure 17: The process of group key update in Panja and proposed scheme

6.1.1. Performance in Two-layer Hierarchical Network

In order to analyze efficiency of SEHKM, it is assumed that all key updates carry out in a two-layer hierarchical network. Let us assume a network $G = \{n_1, n_2, n_3, n_4, \dots, n_N\}$, $n_1, n_2, n_3, n_4, \dots, n_N$ are nodes in the network. The base station is considered as a member of G, ie BS∈G. e_i^{cp} is energy consumption on computation in node n_i and e_i^{cm} is energy consumption on communication in node n_i. Let's assume all clusters have the same size m, ie a cluster has m nodes including the cluster head. The total number of clusters is $(N-1)/m$. The analysis is also based on the following assumptions: all keys have the same size; One encryption or a decryption of the same length of key consumes 1 computation energy unit E; A transmission of the key consumes 1 transmission energy unit T; One key needs 1 storage unit S.

6.1.1.1. The energy consumption on first distribution of all keys

In the last chapter, it is presented that the energy consumption on initial distribution of all keys is different from cost on key update. It assumes $E_{initial}$ is the energy consumption on first distribution of all keys. Then $E_{initial}$ should include two parts: computation and communication costs.

- **Computation cost for cryptography:**

- **Network key:** The key is encrypted by BS at first and the cost is 1(E). The cost for all cluster heads to receive and decrypt the key is $(N-1)/m$ (E). One cluster head encrypts the key and it costs 1(E), then decryption in member nodes cost $m-1$ (E). So the computation cost in one cluster is $1+m-1=m$ (E). There are $(N-1)/m$ clusters in the network, so they totally cost $N-1$ (E). Thus computation cost for network key update is $(N-1)/m + N$ (E).
- **Group key**: First, BS encrypts the key by each pairwise key and each cluster head decrypts the key. It costs $2(N-1)/m$ (E). Then a cluster head encrypts the

key by each pairwise key and each node decrypts the key. It costs $2(m-1)(N-1)/m$ (E) for all clusters. So the total cost is $2m(N-1)/m$ (E).

- **Pairwise key**: First BS encrypts its seed and broadcasts it. Each cluster head receives the seed and decrypts it. It costs $(N-1)/m+1$ (E). Then each cluster head encrypts its seed and BS decrypts all seeds. It costs $2(N-1)/m$ (E). In higher layer, the cost is $3(N-1)/m+1$ (E). In lower layer, the process is similar and the cost is $(3m-2)(N-1)/m$ (E). The keys shared by an assistant node and member nodes cost $2(m-2)(N-1)/m$ (E) and the keys shared by BS and all assistant nodes cost $2(N-1)/m$ (E). The total cost for all pairwise keys is $(5m-1)(N-1)/m+1$ (E)

The total computation cost on initial distribution of all keys is:

$$\sum_{i=1}^{N} e_i^{cp} = 8N + \frac{N-1}{m} - 6 \text{ (E)} \qquad 6.1$$

➢ **Communication cost:**

- **Network key:** BS broadcasts the network key to all cluster heads and then cluster heads broadcasts the key to all member nodes. It costs $1+(N-1)/m$ (T).
- **Group key:** It is similar with network key and it costs $1+(N-1)/m$ (T).
- **Pairwise key:** The seed exchange on the higher layer costs $1+(N-1)/m$ (T) and the lower layer costs $N-1$ (T). The assistant node associated pairwise key costs $N-1-(N-1)/m$ (T). So the total cost on pairwise key transmission is $2N-1$ (T).

Therefore the total communication cost on first distribution of all keys is:

$$\sum_{i=1}^{N} e_i^{cm} = 2N + \frac{2(N-1)}{m} + 1 \text{ (T)} \qquad 6.2$$

The energy consumption on the first distribution of all keys can be shown as:

$$\begin{aligned} E_{initial} &= \sum_{i=1}^{N} \left(e_i^{cp} + e_i^{cm}\right) \\ &= \left(8N + \frac{N-1}{m} - 6\right) \text{(E)} + \left[2N + \frac{2(N-1)}{m} + 1\right] \text{(T)} \end{aligned} \qquad 6.3$$

6.1.1.2. The energy consumption on group key update

In SEHKM, the group key update algorithm is different from the first establishment. Thus energy cost on key update is different too. It assumes E_{update} is the energy consumption on each time of group key update. The computation and communication costs are:

➢ **Computation cost for cryptography:** In higher layer, BS encrypts the key and all cluster heads decrypt the key, it costs $1+(N-1)/m$ (E). In lower layer, a cluster head encrypts the key and all member nodes decrypt the key, all clusters cost $N-1$ (E). So the total cost on computation for cryptography is $N+(N-1)/m$ (E). The total computation cost on group key update is:

6.4

$$\sum_{i=1}^{N} e_i^{cp} = N + \frac{N-1}{m} \text{ (E)}$$

- **Communication cost:** The regular group key update is that cluster head broadcasts the group key in the cluster. So the cost is 1+ (N−1)/m (T). The communication cost on group key update is :

$$\sum_{i=1}^{N} e_i^{cm} = \frac{N-1}{m} + 1 \text{ (T)} \quad 6.5$$

The energy consumption on group key update is:

$$E_{update} = \sum_{i=1}^{N} (e_i^{cp} + e_i^{cm}) = \left(N + \frac{N-1}{m}\right)\text{(E)} + \left(\frac{N-1}{m} + 1\right)\text{(T)} \quad 6.6$$

6.1.1.3. Storage cost

In SEHKM, BS, cluster heads, assistant nodes and normal nodes have different storage cost. The storage cost is shown in Table 9:

Table 9: Storage costs in different nodes

Node	Network key (S)	Group key (S)	Pairwise key (S)	Total (S)
Base station	1	1	(N−1)/m	2+ (N−1)/m
Cluster head	1	2	m−1	m+2
Assistant node	1	1	2	4
Member node	1	1	2	4

The table shows a member node has 4 keys, a cluster head has m+2 keys and the base station has 2+ (N−1)/m keys. The storage cost is reasonable.

6.1.1.4. Efficiency of SEHKM compares with LEAP and Panja

SEHKM is proposed based on LEAP and Panja, the network key update in LEAP and SEHKM are similar while Panja does not support network key. Pairwise key in LEAP and SEHKM are both not updated. The cluster key in LEAP and the group key in Panja are the same as the group key in SEHKM which are all shared by a group of nodes. Therefore the efficiency comparison of the three schemes is related to energy consumption on group key or cluster key update. The computation and communication costs on the update of the group key are analyzed to measure the efficiency of LEAP, Panja and SEHKM.

In LEAP, let d to be the number of neighbors of each node. Each node has a cluster key shared with its neighbors, then computation cost is (d+1)(N−1) (E) and communication cost is N−1 (T). In Panja, each member node sends a partial key to cluster head to compute a group key, then cluster head broadcasts the key. The cost on

computation is $3N+ (N-1)/m -2$ (E) and communication cost is $N+ (N-1)/m$ (T). Table 10 shows the comparison of computation and communication costs on group key update in LEAP, Panja and SEHKM.

Table 10: The efficiency comparison of SEHKM with LEAP and Panja

Scheme	Computation (E)	Communication (T)
LEAP	$(d+1)(N-1)$	$N-1$
Panja	$3N + \frac{N-1}{m} - 2$	$N + \frac{N-1}{m}$
SEHKM	$N + \frac{N-1}{m}$	$\frac{N-1}{m} + 1$

It shows that for each time of group key or cluster key update, the proposed scheme costs the least energy on both computation and communication.

6.1.1.5. Simulation result

To further analyze the efficiency of the proposed scheme, the simulation of three schemes' implementation on a popular MICA2 [67] sensor node in MATLAB is presented. Table 11 shows the total energy consumption (Joule) on group key update when network size grows with the key size is 32 bytes using DES-CBC for encryption. The simulation results in the table show as that SEHKM always consumes less energy than Panja and LEAP in group key update.

Table 11: The energy consumption (Joule) of group key update in SEHKM, LEAP and Panja

N	m	d	LEAP	Panja	SEHKM
100	20	8	0.33264	0.27179	0.026468
200	20	8	0.66864	0.54395	0.050828
300	20	8	1.00464	0.81611	0.075188
400	20	8	1.34064	1.08827	0.099548
500	20	8	1.67664	1.36043	0.123908
600	20	8	2.01264	1.63259	0.148268
700	20	8	2.34864	1.90475	0.172628
800	20	8	2.68464	2.17691	0.196988
900	20	8	3.02064	2.44907	0.221348
1000	20	8	3.69264	2.99339	0.270068

6.1.2. Performance Comparison with Panja in Multiple Layers Hierarchical Networks

The section 6.1.1 presents performance of SEHKM in a two-layer network. But a hierarchical network can be more than two layers, in which SEHKM and Panja schemes can be employed. This section presents the simulation results on energy cost

of Panja and SEHKM in a multiple layers hierarchical network. Figure 18 shows energy cost when a network degree is increasing. In the left one, network size is always around 10^7 and in the right one, number of cluster member is always 10.

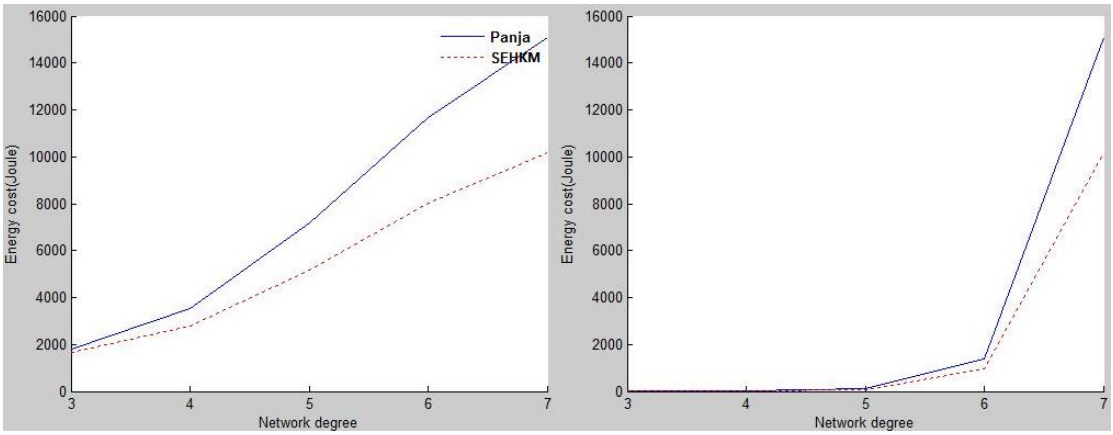

Figure 18: Energy cost when network degree is increasing

6.2. Security and Operation Assessment

The security criteria are proposed in chapter 3, based on the criteria, the security strength of SEHKM can be evaluated. The criteria include keying model, key distribution, key update, node operation and resilience. The proposed scheme is analyzed according to these five aspects.

- **Keying model:** The scheme support network key, group key and pairwise key. Any node in the network has three types of key. The security is very strong.
- **Key distribution:** The distribution of any key in the scheme is secured. In the initialization phase, group key is encrypted by the disposable pairwise key, network key is encrypted by group key and pairwise keying materials are encrypted by other keys, too.
- **Key update:** The proposed scheme supports regular network key and group key update and the specific key update.
- **Node operation:** The scheme supports node addition, revocation and replacement. The keying services for these operations are employed.
- **Resilience:** The SEHKM scheme support securely key update after node compromise.

Compare with LEAP and Panja, SEHKM provides higher security strength as it is shown in Table 12.

The operation assessment of SEHKM is high, too. The hierarchical structure support scalability of a network, the network key ensures the accessibility and the scheme supports node operation which leads to better flexibility.

Table 12: The security comparison of SEHKM with LEAP and Panja

Security criteria	LEAP	Panja	SEHKM
Keying model (20)	All (20)	Group key only (10)	All (20)
Key distribution (20)	Encryption by different key (20)	Basic encryption (5)	Group key update is encrypted by the old key (15)
Key update (20)	Specific update (10)	Specific and regular update (20)	Specific and regular update (20)
Node operation (20)	No addition (10)	All (20)	All (20)
Resilience (20)	Security update (20)	Not good (10)	Security update (20)
Total (100)	80	65	95

6.3. Summary

The proposed SEHKM scheme is evaluated in this chapter. The analysis and simulation results indicate the energy consumption of SEHKM is reasonable and more efficient than LEAP and Panja. The security assessment shows SEHKM provides very strong security and the strength is higher than the two schemes. The operation assessment shows the scheme is able to satisfy the operation requirement of wireless sensor networks.

7. Conclusion

Key management is the essential service in network security. However the schemes in traditional network cannot be employed directly in wireless sensor networks due to constraints caused by their unique characteristics. The demand in security, limitations in hardware and unique characteristics of wireless sensor networks lead to special requirements including security, efficiency and operation aspects, which have to be satisfied in key management schemes in wireless sensor security. Many schemes are proposed for wireless sensor network security. However, there are still potential for further improvement in many aspects. Most existing research aims on security, but it is very hard to have a universal quantized assessment criteria. In the existing LEAP scheme, security is provided by four types of key. But it has problems including lack of flexibility and potential disclosing pairwise keys since a pairwise key is established by the same initial key. Panja scheme employs group Diffie-Hellman algorithm which is able to establish secure group key. However the scheme only supports the group key, if any node is compromised, the whole cluster is in danger.

This research focuses on following issues related to key management in wireless sensor network security:

- How can the criteria be established to assess the security and efficiency of the key management schemes?
- How can the security of other nodes in the same cluster be strengthened when the cluster head is compromised in hierarchical wireless sensor networks?
- How can the existing key update algorithm on hierarchical wireless sensor network be improved on efficiency?

The first research question which is related to the assessment criteria of key management schemes is addressed in chapter 2. All key management schemes have to satisfy security, efficiency and operation requirements of wireless sensor networks, so the assessment should also include all three aspects. Efficiency is easy to evaluate by simulation or analysis while security is always assessed subjectively and it is not easy to do comparison. To address this problem, security criteria are proposed based on processes of key management. Key management schemes' security should be assessed in keying model, key distribution, key update, node operation and resilience. The simplest keying models are network key, group key and pairwise key. Usually a mixed model scheme is more secure than a single model scheme. However a single model scheme which supports pairwise keys also offers strong security. Key distribution should be carried out securely by different keys or using secret seeds. A secure scheme should support both regular key update and specific key update in some situations such as node compromise, node revocation and node replacement. Node operations including addition, revocation and replacement also should be

supported in a key management scheme in wireless sensor network to secure forward and backward secrecy. Resilience against node capture is also essential. The consequence of a node compromise on other nodes in a cluster should be as small as possible.

In hierarchical wireless sensor networks, nodes are divided into clusters. A cluster head can manage and control member nodes in the cluster. The network gets benefits from this structure in data aggregation, less routing, more sleeping time for normal nodes and convenience in network management. Cluster heads are one of the key points in this structure. If a cluster head is compromised, the whole cluster will be in danger. An assistant node management is proposed to address this problem in chapter 4. An assistant node is able to communicate with the head of its cluster head and other member nodes in the cluster. It can arrange other nodes to join other clusters if the cluster head is compromised or damaged.

With the introduction of assistant node, a new key management scheme for cluster based hierarchical wireless sensor network is proposed. The new scheme, SEHKM, has addressed the drawbacks of two existing schemes Panja and LEAP. A concept of initial key is adopted in the new scheme to establish pairwise keys. The update of the initial key has enabled new nodes to securely join the network. A new group key generation algorithm based on pseudo-random numbers and group Deffie-hellman is used. Three types of keys, ie network key, group key and pairwise key are supported in SEHKM. The key generation, distribution and update are fully specified. The analysis and the simulation have shown that the new scheme has improvement on security strength and operation efficiency compared with the two popular schemes Panja and LEAP.

In this research, new assessment criteria for wireless sensor network key management schemes are established. As wireless sensor networks are still evolving structure wise and security mechanism wise, new factors may be added to the assessment criteria. This can be studied in the future. Using Diffie-Hellman algorithm for pairwise key is expensive on communication. An alternative algorithm or new mechanism can be explored in the future.

BIBLIOGRAPHY

[1] J. Zheng and A. Jamalipour, *Wireless Sensor Networks: A Networking Perspective*: A John & Sons, Inc, and IEEE, 2009.

[2] R. Rajagopalan and P. K. Varshney, "Data aggregation techniques in sensor networks: A survey," 2006.

[3] I. F. Akyildiz and M. C. Vuran, *Wireless Sensor Networks*. Singapore: Markono Print Media Pte Ltd, 2010.

[4] C.-Y. Chong, S. Mori, and K.-C. Chang, "Distributed multitarget multisensor tracking," *Multitarget-multisensor tracking: Advanced applications*, vol. 1, pp. 247-295, 1990.

[5] C.-Y. Chong, K.-C. Chang, and S. Mori, "Distributed tracking in distributed sensor networks," in *American Control Conference*, 1986, pp. 1863-1868.

[6] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, *et al.*, "Deploying a wireless sensor network on an active volcano," *IEEE Internet Computing*, vol. 10, pp. 18-25, 2006.

[7] I. F. Akyildiz and M. C. Vuran, *Wireless sensor networks* vol. 4: John Wiley & Sons, 2010.

[8] R. Lin, Z. Wang, and Y. Sun, "Wireless sensor networks solutions for real time monitoring of nuclear power plant," in Proceedings of *the Fifth World Congress on Intelligent Control and Automation, WCICA*, 2004, pp. 3663-3667.

[9] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, pp. 393-422, 3/15/ 2002.

[10] J. A. Mazurek, J. E. Barger, M. Brinn, R. J. Mullen, D. Price, S. E. Ritter *and* D. Schmitt, "Boomerang mobile counter shooter detection system," 2005, pp. 264-282.

[11] T. He, S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J.A. Stankovic, T. F. Abdelzaher, J. Hui and B. Krogh, "VigilNet: An integrated sensor network system for energy-efficient surveillance," *ACM Trans. Sen. Netw.*, vol. 2, pp. 1-38, 2006.

[12] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications, Atlanta, Georgia, USA, 2002.

[13] R. Szewczyk, E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring, and D. Estrin, "Habitat monitoring with sensor networks," *Commun. ACM*, vol. 47, pp. 34-40, 2004.

[14] A. M. Baptista, "CORIE: the first decade of a coastal-margin collaborative observatory," DTIC Document 2006.

[15] D. Malan, T. Fulford-Jones, M. Welsh, and S. Moulton, "Codeblue: An ad hoc sensor network infrastructure for emergency medical care," in *International workshop on wearable and implantable body sensor networks*, 2004.

[16] L. Yong-Min, W. Shu-Ci, and N. Xiao-Hong, "The architecture and characteristics of wireless sensor network," in Proceedings of *International Conference on Computer Technology and Development*, 2009, pp. 561-565.

[17] V. Raghunathan, C. Schurgers, S. Park, and M. B. Srivastava, "Energy-aware wireless microsensor networks," *IEEE Signal Processing Magazine*, vol. 19, pp. 40-50, 2002.

[18] D. W.Carman, P. S. Kruus, and B. J. Matt, *Constraints and Approaches for Distributed Sensor Security*: NAI Labs Tech. Rep.#00-010, 2000.

[19] S. Gajjar, S. Pradhan, and K. Dasgupta, "Wireless sensor network: Application led research perspective," in *Recent Advances in Intelligent Computational Systems (RAICS), IEEE*, 2011, pp. 025-030.

[20] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, "System architecture directions for networked sensors," in *ACM SIGOPS operating systems review*, 2000, pp. 93-104.

[21] C. Haowen, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in Proceedings of *Symposium on Security and Privacy*, 2003, pp. 197-213.

[22] V. Rathod and M. Mehta, "Security in wireless sensor network: a survey," *Ganpat University Journal of Engineering and Technology,* vol. 1, pp. 35-44, 2011.

[23] W. Yong, G. Attebury, and B. Ramamurthy, "A survey of security issues in wireless sensor networks," *Communications Surveys & Tutorials, IEEE,* vol. 8, pp. 2-23, 2006.

[24] X. Chen, K. Makki, K. Yen, and N. Pissinou, "Sensor network security: a survey," *IEEE Communications Surveys & Tutorials*, vol. 11, pp. 52-73, 2009.

[25] G. Dini and M. Tiloca, "Considerations on security in zigbee networks," in Proceeding of *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC),* 2010, pp. 58-65.

[26] A. K. Srivastava and A. Goel, "Security solution for WSN using mobile agent technology," *International Journal of Research and Reviews in Wireless Sensor Networks (IJRRWSN),* vol. 1, 2011, pp. 48-52

[27] J. C. Lee, V. Leung, K. H. Wong, J. Cao, and H. C. Chan, *IEEE Transactions on wireless communications* "Key management issues in wireless sensor networks: current proposals and future developments," vol. 14, pp. 76-84, 2007.

[28] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in the Proceedings of the 9th ACM conference on Computer and communications security, Washington, DC, USA, 2002.

[29] W. Du, J. Deng, Y. S. Han, P. K. Varshney, J. Katz, and A. Khalili, "A pairwise key predistribution scheme for wireless sensor networks," *ACM Transactions on Information and System Security (TISSEC),* vol. 8, pp. 228-258, 2005.

[30] D. Liu, P. Ning, and R. Li, "Establishing pairwise keys in distributed sensor networks," *ACM Transactions on Information and System Security (TISSEC),* vol. 8, pp. 41-77, 2005.

[31] D. Liu and P. Ning, "Location-based pairwise key establishments for static sensor networks," in Proceedings of *the 1st ACM workshop on Security of ad hoc and sensor networks*, 2003, pp. 72-82.

[32] D. Huang, M. Mehta, D. Medhi, and L. Harn, "Location-aware key management scheme for wireless sensor networks," in Proceedings of *the 2nd ACM workshop on Security of ad hoc and sensor networks*, 2004, pp. 29-42.

[33] H. Chan and A. Perrig, "PIKE: Peer intermediaries for key establishment in sensor networks," in Proceedings of *24th Annual Joint Conference of the IEEE Computer and Communications Societies.* 2005, pp. 524-535.

[34] S. Zhu, S. Setia, and S. Jajodia, "LEAP+: Efficient security mechanisms for large-scale distributed sensor networks," *ACM Transactions on Sensor Networks*, vol. 2, pp. 500-528, 2006.

[35] M. Eltoweissy, M. Moharrum, and R. Mukkamala, "Dynamic key management in sensor networks," *IEEE Communications Magazine*,vol. 44, pp. 122-130, 2006.

[36] C. Kyung, Y. Minjung, C. Kijoon, and K. Mihui, "An enhanced key management using ZigBee Pro for wireless sensor networks," in Proceedings of *International Conference on Information Networking (ICOIN)*,2012, pp. 399-403.

[37] R. Blom, "An Optimal Class of Symmetric Key Generation Systems," in *Advances in Cryptology*, pp. 335-338. Springer Berlin Heidelberg, 1985.

[38] M. Steiner, G. Tsudik, and M. Waidner, "Key agreement in dynamic peer groups," *IEEE Transactions on Parallel and Distributed Systems*, vol. 11, pp. 769-780, 2000.

[39] B. Panja, S. K. Madria, and B. Bhargava, "Energy and communication efficient group key management protocol for hierarchical sensor networks," in Proceedings of *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, 2006, p. 8 pp.

[40] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, "SPINS: Security protocols for sensor networks," *Wireless networks*, vol. 8, pp. 521-534, 2002.

[41] J. Jang, T. Kwon, and J. Song, "A Time-Based Key Management Protocol for Wireless Sensor Networks," in *Information Security Practice and Experience* (pp. 314-328). Springer Berlin Heidelberg, 2007.

[42] B. Maala, H. Bettahar, and A. Bouabdallah, "TLA: A Tow Level Architecture for Key Management in Wireless Sensor Networks," in Proceedings of *the 2nd International Conference on Sensor Technologies and Applications,SENSORCOMM '08.*, 2008, pp. 639-644.

[43] D. Brown, "Standards for efficient cryptography, SEC 1: elliptic curve cryptography," *Released Standard Version*, vol. 1, 2009.

[44] Z. Alliance, "Zigbee specification," ed, 2006.

[45] J. Hwang and Y. Kim, "Revisiting random key pre-distribution schemes for wireless sensor networks," in *Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*, 2004, pp. 43-52.

[46] A. Salomaa, *Public-key cryptography*: Springer Science & Business Media, 2013.

[47] D. Davis, "Kerberos plus RSA for world wide web security," in *Proceedings of the 1st USENIX Workshop on Electronic Commerce*, 1995, pp. 185-188.

[48] B. Tung, "Public key cryptography for initial authentication in Kerberos,"*http://www.ietf. org/internet-drafts/draft-ietf-cat-kerberos-pk-init-15. txt* (2001).

[49] H. Alzaid, D. Park, J. Nieto, C. Boyd, and E. Foo, "A Forward and Backward Secure Key Management in Wireless Sensor Networks for PCS/SCADA," in *Sensor Systems and Software* (pp. 66-82). Springer Berlin Heidelberg, 2010.

[50] P. Albers, O. Camp, J.-M. Percher, B. Jouga, L. Mé, and R. S. Puttini, "Security in Ad Hoc Networks: a General Intrusion Detection Architecture Enhancing Trust Based Approaches," in *Wireless Information Systems*, 2002, pp. 1-12.

[51] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Communications of the ACM,* vol. 43, pp. 51-58, 2000.

[52] X. Zhang and J. Wang, "Key Management in Wireless Sensor Networks: Development and Challenges," in *Applied Mechanics and Materials*, 2014, pp. 654-660.

[53] W. Merrill, K. Sohrabi, L. Girod, J. Elson, F. Newberg, and W. J. Kaiser, "Open standard development platforms for distributed sensor networks," in *AeroSense 2002*, pp. 327-337. International Society for Optics and Photonics.

[54] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE Wireless communications,* vol. 11, pp. 6-28, 2004.

[55] H. Nakayama, N. Ansari, A. Jamalipour, and N. Kato, "Fault-resilient sensing in wireless sensor networks," *Computer Communications,* vol. 30, pp. 2375-2384, 2007.

[56] M. Handy, M. Haase, and D. Timmermann, "Low energy adaptive clustering hierarchy with deterministic cluster-head selection," in Proceedings of *the 4th International Workshop on Mobile and Wireless Communications Network.*, 2002, pp. 368-372.

[57] A. Manjeshwar and D. P. Agrawal, "TEEN: a routing protocol for enhanced efficiency in wireless sensor networks," in Proceedings of *the 15th International Symposium on Parallel and Distributed*, 2000, pp. 2009-2015..

[58] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, pp. 366-379, 2004.

[59] H. Chan and A. Perrig, "ACE: An emergent algorithm for highly uniform cluster formation," in *Wireless Sensor Networks* (pp. 154-171). Springer Berlin Heidelberg, 2004.

[60] Q. Fang, F. Zhao, and L. Guibas, "Lightweight sensing and communication protocols for target enumeration and aggregation," in Proceedings of *the 4th ACM international symposium on Mobile ad hoc networking & computing*, 2003, pp. 165-176.

[61] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless

sensor networks," *Computer communications*, vol. 30, pp. 2826-2841, 2007.
[62] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: Modeling and analysis of a three-tier architecture for sparse sensor networks," *Ad Hoc Networks*, vol. 1, pp. 215-233, 2003.
[63] V. Mhatre and C. Rosenberg, "Design guidelines for wireless sensor networks: communication, clustering and aggregation," *Ad hoc networks*, vol. 2, pp. 45-63, 2004.
[64] S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in Proceedings of *the Twenty-Second IEEE Annual Joint Conference on Computer and Communications. IEEE Societies*, 2003, pp. 1713-1723.
[65] S. Bandyopadhyay and E. J. Coyle, "Minimizing communication costs in hierarchically-clustered networks of wireless sensors," *Computer Networks*, vol. 44, pp. 1-16, 2004.
[66] S. William and W. Stallings, *Cryptography and Network Security, 4/E*: Pearson Education India, 2006.
[67] C. Chih-Chun, S. Muftic, and D. J. Nagel, "Measurement of Energy Costs of Security in Wireless Sensor Nodes," in Proceedings of *the 16th International Conference on Computer Communications and Networks (ICCCN)*, 2007, pp. 95-102.